КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

Факультет інформаційних технологій

Кафедра технологій управління

Спеціальність 122 – Комп'ютерні науки,
освітня програма «Інформаційна аналітика та впливи»

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему:

"Інформаційний аналіз та прогнозування даних у сфері E-Commerce"

Студента 2-го курсу групи ІАВ-21

Колумбета Антона Петровича
(прізвище, ім'я, по батькові)

(підпис студента)

Науковий керівник:

д.т.н., професор
(науковий ступінь, вчене звання)

Хлевна Юлія Леонідівна
(прізвище, ім'я, по батькові)

(дата) (підпис)

Попередній захист:

(Висновок: «До захисту в Екзаменаційній комісії»)

Завідувач кафедри
технологій управління

(підпис) (прізвище, ініціали) (дата)

Київ – 2023

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА
Факультет інформаційних технологій

Кафедра технологій управління
Освітньо-кваліфікаційний рівень Магістр
Спеціальність 122 - Комп'ютерні науки
Освітня програма Інформаційна аналітика та впливи

ЗАТВЕРДЖУЮ
Завідувач кафедри
професор Морозов В.В.

_____
«____» _____20___року

З А В Д А Н Н Я
НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Студент Колумбет Антон Петрович

Група ІАВ-21

1. Тема кваліфікаційної роботи
Інформаційний аналіз та прогнозування даних у сфері E-Commerce

Затверджена наказом по від «___»_____20__ р. № ___.

2. Строк подання студентом готової роботи – "___"_____ 20___ р.

3. Цільова установка та вихідні дані до роботи: дослідження використання методів машинного навчання, інформаційного аналізу та прогнозування даних у сфері e-commerce, розробка моделей машинного навчання, опис методики застосування розроблених рішень.

4. Зміст роботи: аналіз предметної області та обгрунтування доцільності застосування методів машинного навчання, інформаційної аналітики та прогнозування для застосування у сфері e-commerce, огляд досвіду підприємств де дані методи застосовуються та тих, де їх можна застосувати, аналіз методів та алгоритмів, аналіз даних, будування моделей, впровадження моделей, розгляд методик використання.
5. Перелік графічного матеріалу (слайдів) Схема життєвого циклу процесу дослідження даних за методологією CRISP-DM, побудова дерева Ball Tree, приклади існуючих CMS, зображення роботи та результатів розроблених методів, веб-сторінка приклад підприємства електронної комерції, приклад використання методу «ліктя» для пошуку кількості кластерів.

6. Календарний план виконання роботи:

№ з/п	Назва частин роботи	%	Виконання роботи	
			За планом	Фактично
1	Вибір теми дипломної роботи	3	01.10.2022	01.10.2022
2	Протокол кафедри ТУ про затвердження тем дипломних робіт та призначення наукових керівників	2	08.12.2022	27.12.2022
3	Формування переліку матеріалів, літератури з проблематики дипломної роботи	10	08.01.2023	08.01.2023
4	Складання розгорнутого плану кваліфікаційної роботи	5	18.01.2023	18.01.2023
5	Ознайомлення наукового керівника з розгорнутим планом кваліфікаційної роботи. Внесення змін.	5	20.01.2023	20.01.2023
6	Підготовка розділу 1	10	13.02.2023	13.02.2023
7	Підготовка розділу 2	14	06.03.2023	06.03.2023
8	Підготовка розділу 3	14	03.04.2023	03.04.2023
9	Підготовка розділу 4	13	17.04.2023	17.04.2023
10	Оформлення кваліфікаційної роботи. Підготовка висновків і пропозицій	15	01.05.2023	01.05.2023
11	Передача кваліфікаційної роботи науковому керівникові	2	02.05.2023	02.05.2023
12	Передача кваліфікаційної роботи рецензенту для рецензування	2	10.05.2023	10.05.2023
13	Попередній захист кваліфікаційної роботи	2	17.05.2023	17.05.2023

Дата видачі завдання «____» _____ 20__ р.

Керівник роботи д.т.н. професор Хлевна Юлія Леонідівна
(посада, прізвище, ім’я, по батькові)

_____
(підпис)

Завдання прийняв до виконання студент групи ІАВ-21

Колумбет Антон Петрович
(прізвище, ім’я, по батькові)

_____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва частин роботи	%	Виконання роботи	
			За планом	Фактично
1.	Вибір теми дипломної роботи	3	01.10.22	01.10.22
2.	Протокол кафедри ТУ про затвердження тем дипломних робіт та призначення наукових керівників	2	08.12.22	27.12.22
3.	Формування переліку нормативних матеріалів, літератури з проблематики дипломної роботи	10	08.01.23	08.01.23
4.	Складання розгорнутого плану кваліфікаційної роботи	5	18.01.23	18.01.23
5.	Ознайомлення наукового керівника з розгорнутим планом кваліфікаційної роботи. Внесення змін.	5	20.01.23	20.01.23
6.	Підготовка розділу 1	10	13.02.23	13.02.23
7.	Підготовка розділу 2	14	06.03.23	06.03.23
8.	Підготовка розділу 3	14	03.04.23	03.04.23
9.	Підготовка розділу 4	13	17.04.23	17.04.23
10.	Оформлення кваліфікаційної роботи. Підготовка висновків і пропозицій	15	01.05.23	01.05.23
11.	Передача кваліфікаційної роботи науковому керівникові	2	02.05.23	02.05.23
12.	Передача кваліфікаційної роботи рецензенту для рецензування	2	10.05.23	10.05.23
13.	Попередній захист кваліфікаційної роботи	5	17.05.23	17.05.23

ЗМІСТ

АНОТАЦІЯ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
Факультет інформаційних технологій
Кафедра технологій управління
Спеціальність 122 - Комп'ютерні науки,
освітня програма "Інформаційна аналітика та впливи"

Дипломна робота магістра Колумбета Антона Петровича.

Тема роботи – «Інформаційний аналіз та прогнозування даних у сфері E-Commerce».

Мета кваліфікаційної роботи магістра – розробити методику для застосування інформаційного аналізу та прогнозування даних сфери електронної діяльності, з використанням методів Machine Learning.

Об'єкт дослідження – процеси електронної комерції в діяльності підприємств.

Предмет дослідження – методики застосування машинного навчання для інформаційного аналізу та прогнозування.

Наукова новизна роботи – розроблено методику застосування інформаційного аналізу та прогнозування в бізнесовій діяльності компанії, яка відрізняється від існуючих тим, що надає вагому кількість інформації з використанням машинного навчання, яка може бути застосована в діяльності електронної комерції, а саме маркетингових підходах upsell та cross-sell та надає інформацію про наступну покупку клієнтів ще на стадії заповнення кошику.

У роботі досліджуються існуючі підходи до інформаційного аналізу та прогнозування у задачах керування проектною діяльністю компанії. Розробляється методика їх використання, а також проводиться обґрунтування доцільності та необхідності впровадження запропонованої методики. Наводяться рекомендації щодо практичної імплементації методики.

Дипломна робота складається зі вступу, основної частини, яка включає чотири розділи, висновків та списку використаних джерел. Всього налічує 95 сторінок та перелік посилань з 40 джерел на 11 сторінках.

Ключові слова: електронна комерція, інформаційний аналіз даних, машинне навчання, методика, метод, методологія, прогнозування.

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ

Скорочення

ЕК – Електронна комерція

CMS - Content Management System (Система управління контентом)

CRM - Customer Relationship Management (Управління взаємовідносинами з клієнтами)

Терміни

Датасет – набір даних.

Електронна комерція – це сфера цифрової економіки, що включає всі фінансові та торгові транзакції, які проводяться за допомогою комп'ютерних мереж

Data Science – це міждисциплінарна галузь про наукові методи, процеси і системи, які стосуються добування знань із даних у різних формах, як структурованих так і неструктурованих.

CMS - Це програмне забезпечення, яке дозволяє створювати, редагувати та керувати вмістом веб-сайту без необхідності глибоких знань веб-програмування. CMS надає інтерфейс користувача, який спрощує процес створення та оновлення веб-сторінок, дозволяє керувати медіа-файлами, створювати і редагувати структуру сайту та керувати правами доступу до контенту.

CRM - Це підхід та стратегія управління взаємодією з клієнтами з метою забезпечення їх задоволеності та побудови довгострокових взаємовигідних відносин. CRM може включати в себе використання спеціалізованого програмного забезпечення (CRM-системи), яке дозволяє збирати, аналізувати та керувати інформацією про клієнтів, відстежувати взаємодію з ними, керувати продажами та послугами, розвивати маркетингові кампанії та покращувати загальний досвід клієнтів.

ВСТУП

Використання Data Science, аналізу та прогнозування даних у сфері E-Commerce є ключовим фактором успіху для підприємств у сучасному цифровому віку. Зростання електронної комерції в останні роки створило безліч можливостей, але також поставило перед компаніями великий обсяг даних, які необхідно ефективно аналізувати та використовувати для прийняття стратегічних рішень.

Data Science, як інтердисциплінарна галузь, поєднує статистику, математику, комп'ютерні науки та експертний досвід з метою отримання цінної інформації з великих обсягів даних. У контексті E-Commerce, аналіз та прогнозування даних стають невід'ємною частиною стратегічного управління, маркетингу та оперативної діяльності підприємств.

Застосування Data Science у сфері E-Commerce дозволяє компаніям здійснювати персоналізацію пропозицій, прогнозувати попит, оптимізувати ціноутворення та запаси товарів, аналізувати поведінку покупців та ефективність маркетингових кампаній. Велика кількість даних, що накопичується через онлайн-торгівлю, надає можливості для виявлення нових трендів, виявлення сегментів клієнтів та розробки персоналізованих рекомендацій.

Більш точне розуміння покупців і їх потреб дозволяє підприємствам покращувати якість обслуговування, пропонувати кращі товари та послуги, збільшувати витрати покупців та покращувати загальну задоволеність клієнтів. Аналіз та прогнозування даних також допомагають виявляти причинно-наслідкові зв'язки між різними факторами, такими як ціни, акції, маркетингові кампанії тощо, та їх вплив на продажі та дохід. Це дає змогу підприємствам вдосконалювати свої стратегії, вирішувати проблеми та впроваджувати інновації для забезпечення стабільного росту та конкурентоспроможності.

Окрім аналізу внутрішніх даних, Data Science дозволяє використовувати зовнішні дані, такі як соціальні медіа, рейтинги, відгуки клієнтів тощо, для

отримання більш повного розуміння ринкових тенденцій та конкурентного середовища. Це допомагає підприємствам приймати обґрунтовані рішення щодо рекламних кампаній, стратегій залучення клієнтів та розвитку нових продуктів.

Застосування Data Science в сфері E-Commerce має значний потенціал для підвищення ефективності та досягнення конкурентної переваги. Відповідне використання аналізу даних та прогнозування дозволяє підприємствам розуміти своїх клієнтів, оптимізувати свої процеси та приймати обґрунтовані стратегічні рішення. Це стає ключовим фактором успіху в конкурентному світі електронної комерції, де швидкість, точність та розуміння даних мають вирішальне значення. конверсії. Застосування методів машинного навчання дозволяє виявити сегменти клієнтів, їхні інтереси та потреби, що в свою чергу допомагає розробляти цілеспрямовані маркетингові стратегії для кожного сегменту.

Data Science також може покращити процеси логістики та управління ланцюгом постачання в електронній комерції. Аналізуючи дані про постачальників, транспортування та складські запаси, можна оптимізувати маршрутизацію, розподіл ресурсів та зменшити затрати. Таким чином, підприємства можуть досягти високої ефективності упровадження та доставки товарів, що сприяє підвищенню задоволення клієнтів та забезпечує конкурентну перевагу.

У сучасному електронному бізнесі, де обсяги даних постійно зростають, використання Data Science, аналізу та прогнозування даних стає необхідністю. Вони дозволяють підприємствам отримати цінну інформацію, приймати обґрунтовані рішення та вигравати в конкурентній боротьбі. Аналіз даних та прогнозування у сфері E-Commerce забезпечує підприємствам можливість вдосконалювати свою стратегію, підвищувати прибутковість та надавати більш персоналізований та задоволений клієнтами досвід покупок.

Мета роботи – розробити методику для застосування інформаційного аналізу та прогнозування даних сфери електронної діяльності, з використанням методів Machine Learning для добування корисних даних, та отримання, з їх

допомогою, математичних та програмно реалізованих результатів, завдяки яким можна закріпити, поглибити та узагальнити теоретичні й практичній знання в інформаційному аналізі та прогнозуванні даних.

Об’єктом дослідження процеси електронної комерції в діяльності підприємств.

Предметом дослідження магістерської роботи є методики застосування машинного навчання для інформаційного аналізу та прогнозування.

В якості інформаційних джерел було розглянуто наукові роботи авторів зі всього світу, статті, документація до технологій, програмного забезпечення, та бібліотек.

Наукова новизна роботи. розроблено методику застосування інформаційного аналізу та прогнозування в бізнесовій діяльності компанії, яка відрізняється від існуючих тим, що надає вагому кількість інформації з використанням машинного навчання, яка може бути застосована в діяльності електронної комерції, а саме маркетингових підходах upsell та cross-sell та надає інформацію про наступну покупку клієнтів ще на стадії заповнення кошику.

Для задоволення мети роботи необхідно було вирішити такі завдання:

Необхідно застосувати методи Data Science в сфері Електронної комерції для інформаційного аналізу та прогнозування. Слід виокремити переваги та недоліки методів, охарактеризувати модель та алгоритм, мову програмування реалізації методів, формалізувати та оглянути результати.

В якості засобів для розробки були використані сучасні методи та інструменти, які дозволяють аналізувати та візуалізувати дані, та їхня документація: Python, JupyterNotebook та оточення ANACONDA для створення зручного інтерфейсу для роботи з Python та відображення результатів, різноманітні бібліотеки для Python, що допомагають працювати з наборами даних, такі як Pandas, Seaborn, numpy, а також безкоштовний репозиторій з наборами даних Kaggle.

Практичне значення одержаних результатів – Було розроблено моделі інформаційного аналізу даних та прогнозування у сфері електронної комерції. Вони можуть бути використані окремо або в поєднанні з іншими методами для досягнення кращих результатів. Вибір конкретного методу залежить від задачі, типу даних та інших факторів. Засвоєння цих методів дозволяє нам розширити наші знання та навички в галузі машинного навчання та аналізу даних і створити ефективні моделі для розв'язання різноманітних завдань. Було описано методику застосування даних отриманих з роботи моделей, що були розроблені.

Апробація роботи. Результати дослідження роботи методу машинного навчання «Навчання асоціативним правилам» були оприлюдненні у матеріалах конференції «Інформаційні технології та впровадження» 2021 року у роботі Колумбет А., Хлевний А. «Recommender Systems for E-Commerce Using Association Rule Learning» [1].

Кваліфікаційна робота магістра складається зі вступу, чотирьох розділів, висновку та трьох додатків.

РОЗДІЛ 1. АНАЛІЗ ТЕОРЕТИКО-МЕТОДОЛОГІЧНИХ ОСНОВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНОЇ АНАЛІТИКИ ТА ПРОГНОЗУВАННЯ В СФЕРІ E-COMMERCE

1.1. Аналіз сфери продажів електронної комерції

Глобалізація, що знайшла свій прояв і в бізнесі, і в освіті, і в медицині, і в соціальному житті, надала діяльності аналітиків даних значущості. Сьогодні жодна сфера людської діяльності не обходиться без використання методів Data Science. Ставити на власне почуття та інтуїцію – моветон в сучасному розумінні розвитку бізнесу, адже ніщо окрім правильного так ретельного аналізу статистики не спрогнозує чи не підкаже кращого шляху для покращення прибутків підприємств.

Електронна комерція (електронна торгівля) - це купівля та продаж товарів та послуг, передача коштів або даних через електронну мережу, насамперед Інтернет. Ці ділові операції відбуваються як бізнес для бізнесу (B2B), бізнес для споживача (B2C), споживач для споживача або споживач для бізнесу. Терміни "електронна комерція" та "електронний бізнес" часто використовуються як взаємозамінні. Термін "електронна торгівля" також іноді використовується щодо транзакційних процесів, що становлять роздрібну торгівлю через Інтернет.

В своїй роботі Джуда Філліпс писав про те, що він очікує, що світовий ринок електронної комерції зростатиме із середньорічним темпом зростання 17%. Він пише, що у США у третьому кварталі 2015 року обсяг електронної комерції склав 87,5 мільярда доларів та склав 7,4% всіх роздрібних продажів (Rogers 2015). З 2014 року електронна комерція зростає в середньому на 14% - 15% на рік, тоді як зростання роздрібної торгівлі залишалося менше ніж 3%. comScore підрахував, що американські споживачі витратили понад 57 мільярдів доларів в Інтернеті з листопада до грудня 2015 року. Китайська компанія Alibaba повідомила про те, що в один із популярних торгових днів обсяг "товарів" склав понад 14,92 млрд. дол. [1]

Якщо раніше моніторинг продажів відбувався за допомогою використання людських сил – писався практично від руки, то на сьогодні цей спосіб ведення даних можна порівняти з витісуванням слів на кам’яних плитах – в сьогоденні для таких речей існує широкий спектр автоматизованих інструментів.

Зараз для полегшення роботи аналітика в сферах електронної комерції може послугувати велика кількість різноманітних систем керування вмістом (CMS).

CMS - це програмне забезпечення, яке дозволяє створювати, редагувати та публікувати контент. Якщо перші CMS використовувалися для керування документами та локальними комп'ютерними файлами, то зараз більшість систем CMS розроблені виключно для керування вмістом в Інтернеті, поміж іншого дане програмне забезпечення включає в себе різноманітні модулі для ведення статистики продажів, відвідування сайту, збору інформації про клієнтів та інше.

Існує кілька веб-інструментів CMS. Нижче перераховані найбільш популярні з них:

WordPress – безкоштовне програмне забезпечення, призначене для створення веб-сайтів або блогів на основі шаблонів.

Blogger – інструмент Google для ведення блогів, розроблений спеціально для ведення блогу

Joomla – гнучкий інструмент веб-публікації, що підтримує бази даних користувача та розширення

Drupal – платформа з відкритим вихідним кодом, що часто використовується для розробки сайтів на основі спільнот

Weebly – веб-платформа для створення простих персональних та бізнес-сайтів

Wix – набір інструментів веб-публікації для створення високонастроюваних веб-сайтів.

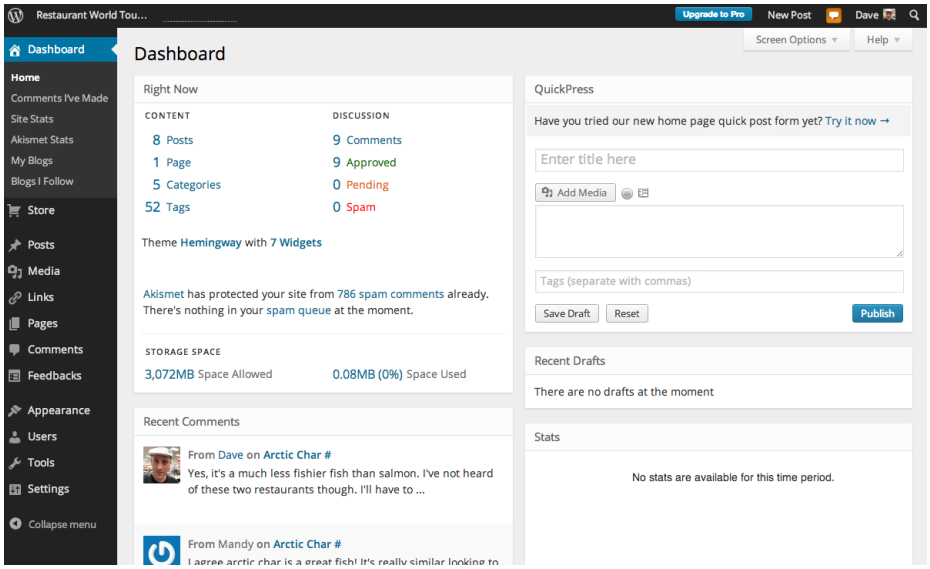


Рисунок 1.1. Зовнішній вигляд інтерфейсу CMS WordPress

Також для моніторингу статистики продаж, клієнтів та іншої корисної інформації може прийти в нагоду CRM – це абревіатура від Customer Relationship Management. Це система, що використовується для побудови та управління взаємовідносинами з клієнтами.

CRM-система допомагає компанії керувати всіма взаємодіями з клієнтами та потенційними клієнтами. CRM-платформа дозволяє фіксувати переваги клієнтів та відслідковувати їхню активність. Таким чином, щоразу, коли з ними розмовляють, незалежно від того, з ким вони спілкуються, клієнти набувають повністю персоналізованого та послідовного досвіду.

Програмне забезпечення CRM допомагає організаціям оптимізувати свої процеси та робочі процеси, щоб усі частини бізнесу були на одній хвилі. Особливо команди з продажу та маркетингу покладаються на CRM для створення спільної роботи та підвищення продуктивності.

Загальна мета полягає в тому, щоб забезпечити більш привабливий досвід роботи з клієнтами, підвищити лояльність та утримати клієнтів, а також стимулювати зростання та прибутковість бізнесу.

Рисунок 1.2. Інтерфейс для взаємодії з CMS iiko

Ще одним за найпопулярніших способів моніторингу та збору статистики є Google Analytics. Але що означає Google Analytics?

Google Analytics - це програма для аналізу відвідуваності сайту, яка надає статистику та аналіз взаємодії користувачів із сайтом у режимі реального часу. Google Analytics дозволяє власникам сайтів аналізувати своїх відвідувачів з метою інтерпретації та оптимізації роботи сайту. Google analytics може відстежувати всі форми цифрових медіа та посилатися на висхідні веб-напрямки, банерну та контекстну рекламу, електронну пошту та інтегрується з іншими продуктами Google. [2]

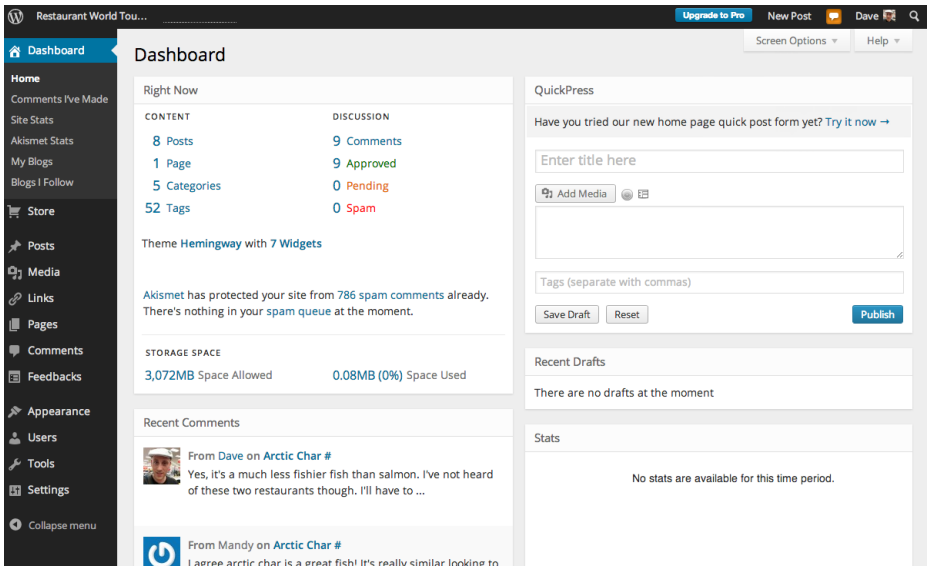


Рисунок 1.3. Інтерфейс для взаємодії з Google Analytics

Дані, що надаються Google Analytics, розроблені спеціально для маркетологів та веб-майстрів, щоб оцінити якість трафіку, який вони отримують, та ефективність їх маркетингових зусиль, також можуть надати багато цікавих даних для досліджень засобами Data Science.

Аналітика Google дозволяє отримати інформацію про результати маркетингової кампанії, відстежуючи відвідувачів з усіх сайтів, що посилаються, і кількість відвідувачів, конвертованих в клієнтів або членів клубу з кожного з них. Аналітика Google працює через фрагмент Javascript на сайті, за яким ведеться спостереження. Немає необхідності встановлювати апаратне або програмне забезпечення, оскільки програма повністю базується на хмарних технологіях.

1.2. Огляд підприємств, для яких є актуальною розробка систем інформаційного аналізу та прогнозування

Загалом інформаційний аналіз та прогнозування є актуальними для підприємств електронної комерції будь-яких масштабів. Це зумовлено тим, що для малих підприємств постійна робота живого аналітика – є занадто дорогоцінним задоволенням, та й для великих підприємств, які постійно прагнуть до мінімізації витрат – це також марне задоволення. В сьогоденні майже всі великі центри електронної комерції мають в своєму використанні системи інформаційного аналізу та прогнозування: Rozetka, Prom.ua та ін., тому слід розглядати менші підприємства, наприклад, гарним прикладом може бути обласний магазин меблів чи одягу.

У сфері електронної комерції розробка систем інформаційного аналізу та прогнозування є критично важливою для багатьох підприємств. Ось огляд деяких типових підприємств, для яких ці системи є актуальними, з конкретними прикладами:

1. Онлайн-роздрібні магазини: Підприємства, які продають товари через Інтернет, використовують системи аналізу та прогнозування для розуміння споживчих тенденцій, аналізу конкуренції та оптимізації цінової політики. Наприклад, Amazon використовує аналітичні системи для персоналізації рекомендацій покупцям, а також для планування запасів та оптимізації доставки.
2. Платіжні системи: Компанії, які надають послуги оплати в електронній комерції, використовують системи аналізу для виявлення шахрайських дій, моніторингу транзакцій та виявлення патернів покупок. Наприклад, PayPal використовує аналітичні системи для виявлення підозрілих транзакцій та захисту від шахрайства.
3. Маркетингові агентства: Агентства, що спеціалізуються на цифровому маркетингу, використовують системи аналізу та прогнозування для визначення ефективності кампаній, аналізу конверсій та розуміння поведінки клієнтів. Наприклад, Google Analytics надає інструменти для аналізу веб-трафіку та ефективності рекламних кампаній.
4. Логістичні компанії: Постачальники логістичних послуг у сфері електронної комерції використовують системи аналізу для оптимізації маршрутів доставки, управління запасами та прогнозування попиту. Наприклад, FedEx використовує аналітичні інструменти для вдосконалення своїх логістичних процесів та планування маршрутів доставки.

Розглянемо підприємство – dybok.com.ua. Для даного підприємства могла б знадобитися система інформаційного аналізу та прогнозування. В сфері продажу меблів може бути корисним система, що підкаже, яка шафа для вітальні краще підійде до нещодавно придбаної кухні, та надасть цю інформацію у відділ продажу для подальшої побудови маркетингової компанії.

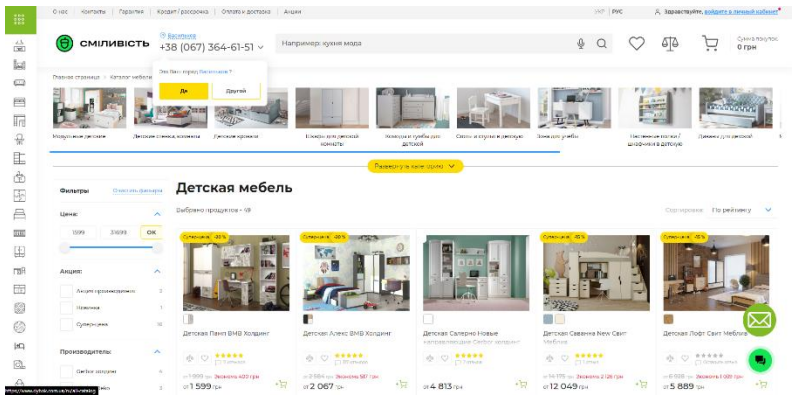


Рисунок 1.4. Титульна сторінка dybok.com.ua

Розробка системи інформаційного аналізу та прогнозування для даного підприємства могла б автоматизувати багато людської роботи та значно покращити його прибутки та продуктивність.

1.3. Поняття Cross-sell та Upsell у сфері електронної комерції

Для розуміння мети розробки моделей машинного навчання необхідно розглянути такі поняття, як Cross-sell та Upsell в електронній комерції.

Upsell - це стратегія маркетингу та продажу, яка полягає у пропозиції клієнту товарів або послуг вищої цінової категорії або більш розширених версій того, що він вже збирався придбати. Основна мета upsell - збільшити суму продажу, пропонуючи клієнту більш дорогі або розширені варіанти продукту, який він вже має намір придбати.

Стратегія upsell може використовуватись в різних сферах бізнесу, включаючи роздрібну торгівлю, готельну галузь, послуги, програмне забезпечення та багато інших. Наприклад, в роздрібній торгівлі продавець може запропонувати клієнту вищий клас товару з більшими функціональними можливостями або додатковими аксесуарами за додаткову плату.

Для успішної реалізації upsell важливо мати глибоке розуміння потреб та бажань клієнта. Продавець повинен вміти ефективно комунікувати переваги та

додаткові можливості продукту вищої цінової категорії, які можуть зацікавити клієнта і змусити його зробити більш вигідний вибір.

Важливо пам'ятати, що стратегія upsell повинна бути зорієнтована на користь клієнта і не має нав'язувати йому непотрібні або неправильні рішення. Продавець повинен добре аргументувати, чому товар або послуга вищої цінової категорії може принести більше задоволення, ефективності або комфорту клієнту.

Успішна реалізація upsell може призвести до збільшення середнього чеку, покращення прибутковості та покращення задоволення клієнтів, які отримують більш високоякісні або розширені продукти.

Cross-sell - це стратегія маркетингу та продажу, яка полягає у пропозиції додаткових товарів або послуг клієнту під час його покупки основного товару або послуги. Це означає, що продавець або компанія стараються збільшити свій дохід, пропонуючи клієнту додаткові продукти, які доповнюють або покращують його основну покупку.

Cross-sell відрізняється від upsell, де фокус звертається на пропозицію більш дорогих або вищого класу товарів замість додаткових. Cross-sell ставить за мету просто поширити покупку клієнта, пропонуючи йому додаткові продукти, які можуть бути взаємопов'язані, суміжні або доповнюють його основний вибір.

Ця стратегія використовується в різних галузях бізнесу, включаючи роздрібну торгівлю, електронну комерцію та фінансові послуги. Наприклад, у роздрібній торгівлі, коли клієнт купує певний товар, продавець може запропонувати йому додаткові аксесуари або супутні товари, які покращують його досвід або задовольняють його потреби.

Для успішної реалізації cross-sell важливо мати розуміння клієнтів і їхніх потреб. Аналіз даних про покупців, їхні звички, покупки та інші показники може допомогти визначити, які додаткові продукти можуть бути цікавими для клієнтів, які купують певний основний товар або послугу.

Ефективна стратегія cross-sell може призвести до збільшення середнього чеку, покращення клієнтського досвіду і підвищення загального обсягу продажів компанії. Проте, важливо пам'ятати, що cross-sell повинен бути здійснений з урахуванням потреб і інтересів клієнтів, щоб уникнути враження нав'язливості або неприємностей.

1.4. Аналіз методологій для прикладних задач інформаційної аналітики та прогнозування

В процесі дослідження даних дослідники часто користуються різномантіними методологіями, для порівняння розглянемо дві популярні методології Data Science – CRISP-DM та BABOK.

CRISP-DM (Cross-Industry Standard Process for Data Mining) — це найбільш поширена практично методологія виконання Data Science проектів, яку прийнято називати міжгалузевим стандартним процесом дослідження даних. Він описує життєвий цикл Data Science проектів у наступних 6 фазах, кожна з яких включає низку завдань [3]:

– розуміння бізнесу (Business Understanding), де через оцінку поточної ситуації визначаються бізнес-мети та вимоги, а також розробляється попередній план проекту;

– початкове вивчення даних (Data Understanding), включаючи їх збирання, опис, дослідження (пошук закономірностей, формування гіпотез) та перевірку якості;

– підготовка даних (Data Preparation), коли з вихідного набору даних формується датасет до роботи з моделями машинного навчання (Machine Learning) шляхом виконання відповідних операцій

– Data Preparation – вибірка очищення, генерація ознак, інтеграція, форматування.

– моделювання (Modeling), де вибираються алгоритми, пишуться тести, будуються та навчаються моделі Machine Learning, якщо вони присутні, а також виконується налаштування їх параметрів та оцінка якості;

– оцінка рішення (Solution Evaluation), коли якість моделей аналізується з точки зору досягнення поставлених бізнес-цілей та визначаються подальші кроки щодо покращення результатів;

– впровадження (Deployment), що передбачає розгортання отриманих моделей у промислову експлуатацію (production), включаючи розробку фінальних звітів у всьому проекті (review).

Рисунок 1.5. Життєвий цикл методології CRISP-DM

BABOK описує набір завдань професійної діяльності бізнес-аналітика, структуруючи їх по 6 галузях знань:

– Планування та моніторинг бізнес-аналізу (Business Analysis Planning and Monitoring)

– Виявлення та співпраця (Elicitation and Collaboration) із зацікавленими сторонами (стейкхолдерами)

– Управління життєвим циклом вимог (Requirements Life Cycle Management)

– Аналіз стратегії (Strategy Analysis)

– Аналіз вимог та визначення дизайнів (Requirements Analysis and Design Definition)

– Оцінка рішення (Solution Evaluation), [19]

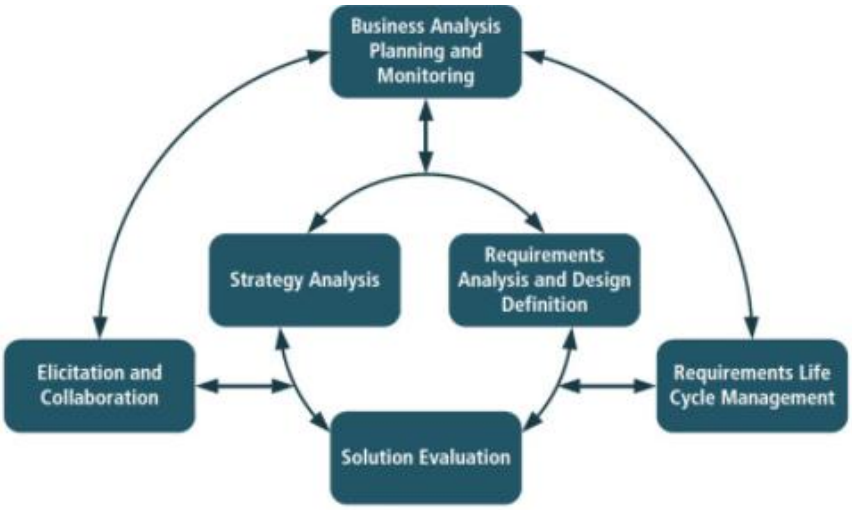


Рисунок 1.6. Життєвий цикл методології BABOK

Загалом, для цілей прикладних задач Data Science в сфері електронної комерції можуть підійти обидва підходи, але більш популярним та широко використовуваним у світі є саме методологія Data science під назвою CRISP-DM.

Для підтвердження цього можна поглянути на дослідження про використання методологій в сфері Data Science. За опитуванням наукового порталу про технології штучного інтелекту, науку про дані та машинне навчання kdnuggets - майже 50% дослідників користуються методологією CRISP-DM (рис 1.6) [15].

Наука даних вимагає низхідного, орієнтованого вирішення проблем підходу. Згідно з останнім дослідженням з підбору персоналу в області Data Science, кількість відкритих вакансій в області Data Science та Analytics досягла максимуму в лютому-березні 2020 року, склавши близько 113 000 в перший тиждень березня і постійно збільшуючись порівняно з 97 000 минулого року. [17]

У цій галузі дані грають дуже важливу роль, а такі процеси, як видобуток даних, допомагають генерувати дієві ідеї, отримувати закономірності та виявляти взаємозв'язки з великих масивів даних. CRISP-DM розроблений таким чином, що не залежить від конкретної області та широко використовується в промисловості та дослідницьких спільнотах.

Завдяки своїм відмінним характеристикам CRISP-DM вважається "де-факто" стандартом методології видобутку даних та еталонною основою, з якою порівнюються інші методології. Одним з важливих факторів використання цього методу Data Science є те, що це міжгалузевий стандарт, який може бути впроваджений в будь-який проект Data Science незалежно від його області. Саме тому для виконання роботи буде використовуватися ця методологія.

1.5. Методики застосування інформаційного аналізу та прогнозування у сфері електронної комерції

Застосування інформаційного аналізу та прогнозування в сфері електронної комерції відіграє вирішальну роль у досягненні успіху та стійкого розвитку підприємств. Ці інструменти надають організаціям можливість зрозуміти своїх клієнтів, оптимізувати бізнес-процеси, покращити маркетингові стратегії та прийняти обґрунтовані рішення.

У сфері електронної комерції інформаційний аналіз та прогнозування використовуються для отримання цінної інформації, покращення прийняття рішень та оптимізації бізнес-процесів. Деякі методики застосування інформаційного аналізу та прогнозування у сфері електронної комерції включають:

Аналіз даних покупців: Збір та аналіз даних про покупців, таких як історія покупок, звички, переваги, рейтинги та відгуки, дозволяє підприємствам розуміти поведінку та потреби клієнтів. Це допомагає покращити персоналізацію, рекомендації товарів та планування маркетингових кампаній.

Прогнозування попиту: Використання статистичних моделей та алгоритмів машинного навчання для прогнозування попиту на товари та послуги. Це дозволяє підприємствам здійснювати ефективне управління запасами, планування виробництва та оптимізацію поставок.

Аналіз конкурентного середовища: Використання інформаційних інструментів для моніторингу конкурентів, аналізу їхньої стратегії

ціноутворення, маркетингових акцій та інших аспектів. Це допомагає підприємствам визначати свою конкурентну позицію та приймати рішення щодо стратегії розвитку.

Аналіз ефективності маркетингових кампаній: Вимірювання та аналіз результатів маркетингових акцій, включаючи ефективність рекламних каналів, конверсію, ROI (прибутковість інвестицій у маркетинг) та інші показники. Це дозволяє підприємствам визначати успішні канали просування та вдосконалювати маркетингові стратегії.

Класифікація та сегментація клієнтів: Використання алгоритмів машинного навчання для класифікації та сегментації клієнтів на основі їхніх характеристик та поведінки. Це дозволяє підприємствам зрозуміти свою аудиторію, налаштувати спеціальні пропозиції та персоналізовані комунікації.

Ці методики інформаційного аналізу та прогнозування допомагають підприємствам у сфері електронної комерції розуміти ринкові тенденції, виявляти можливості та приймати обґрунтовані рішення для покращення своєї продуктивності та конкурентоспроможності.

1.6. Аналіз необхідності застосування інформаційного аналізу та прогнозування у сфері електронної комерції

В сфері електронної комерції інформаційний аналіз та прогнозування необхідні для глибокого розуміння клієнтів, виявлення їх потреб та уподобань, а також для визначення ефективних маркетингових стратегій і оптимізації бізнес-процесів. Ці інструменти дозволяють підприємствам приймати обґрунтовані рішення, прогнозувати попит на товари та послуги, а також забезпечувати персоналізовані пропозиції для клієнтів, що призводить до підвищення задоволення клієнтів та покращення результатів бізнесу.

У процесі аналізу науково-інформаційних джерел на тему використання аналізу та прогнозування даних у сфері електронної комерції (E-commerce), було зроблено висновок, що ця область має великий потенціал і привертає

значну увагу дослідників та практиків. Основні напрямки досліджень включають прогнозування попиту, поведінки клієнтів, персоналізацію рекомендацій, оптимізацію ціноутворення та управління запасами.

У дослідженні "Predictive Analytics in E-commerce: A Review and Future Directions" [1] визначено різні методи прогнозування, включаючи статистичні моделі, машинне навчання, нейромережі та аналіз великих обсягів даних. Автори виявили, що дослідники активно застосовують алгоритми класифікації, регресії, кластеризації та асоціативних правил для прогнозування поведінки клієнтів, рекомендацій товарів та інших факторів, що впливають на результативність електронної комерції.

Інше дослідження "Data Analytics for E-commerce: A Comprehensive Review" [2] визначає ключові аспекти аналітики даних у сфері E-commerce. Автори вказують, що аналітика даних може бути використана для розуміння поведінки клієнтів, виявлення сегментів клієнтів, прогнозування попиту, управління запасами та оптимізації процесів доставки. Вони також наголошують на важливості використання великих обсягів даних (Big Data) та алгоритмів машинного навчання для отримання цінних даних та покращення прийняття рішень у сфері електронної комерції.

Інше дослідження, опубліковане у статті "Predictive Analytics in E-commerce: State-of-the-Art Review and Research Opportunities" [3], висвітлює сучасний стан застосування прогнозування та аналізу даних у сфері E-commerce. Дослідники виявили, що використання прогнозування та аналітики даних може допомогти покращити різні аспекти електронної комерції, такі як управління запасами, планування рекламних кампаній, оптимізація ціноутворення та визначення стратегій залучення та утримання клієнтів.

Дослідження також вказує на можливості використання передових технологій, таких як штучний інтелект, машинне навчання та глибинне навчання, для розв'язання складних проблем електронної комерції. Автори статті підкреслюють значення використання інтегрованих аналітичних систем,

які поєднують дані з різних джерел та застосовують різні аналітичні методи для отримання більш точних та деталізованих результатів.

Загалом, аналіз науково-інформаційних джерел свідчить про важливість та потенціал використання аналізу та прогнозування даних у сфері E-commerce. Застосування цих методів дозволяє покращити стратегічне та оперативне прийняття рішень, підвищити ефективність електронної комерції та забезпечити більш персоналізований підхід до клієнтів. Компанія Amazon, яка вважається однією з провідних у сфері електронної комерції, вдало використовує аналіз та прогнозування даних для покращення свого бізнесу. Вони використовують методи машинного навчання та аналізу великих обсягів даних для розуміння поведінки клієнтів, персоналізації рекомендацій, прогнозування попиту та оптимізації процесів доставки. Це дозволяє їм створити кращий користувацький досвід, забезпечити швидку доставку та максимізувати задоволення клієнтів.

Також варто враховувати випадки успіху компаній, які використовують аналітику даних для зростання своєї електронної комерції. Наприклад, компанія Netflix використовує аналіз даних для персоналізації рекомендацій відеоконтенту своїм користувачам, що дозволяє їм надавати унікальний досвід перегляду та привертати нову аудиторію.

Одним з важливих аспектів використання аналізу та прогнозування даних у сфері E-commerce є захист від шахрайства та кіберзлочинності. Компанії активно використовують аналітичні методи для виявлення підозрілих замовлень, фродулентних дій та зловживань, що дозволяє їм запобігати фінансовим збиткам та забезпечувати безпеку своїм клієнтам. Використання аналітики даних дозволяє виявляти аномальні патерни, розпізнавати зловмисну активність та реагувати на неї в реальному часі.

Дослідження підтверджують, що ефективне використання аналізу та прогнозування даних може значно поліпшити управління запасами та ланцюгом постачання в сфері E-commerce. Аналітичні моделі можуть

прогнозувати попит на товари, виявляти тренди та сезонні зміни, що дозволяє підприємствам забезпечувати наявність потрібних товарів у відповідний час.

Завдяки аналізу даних, підприємства можуть покращити планування запасів, зменшити ризики пов'язані з недостачею або перевищенням товарів. Наприклад, засновуючись на історичних даних про попит на певний товар, аналітичні моделі можуть розробити прогноз на майбутній попит та відповідно запланувати належний рівень запасів. Це дозволяє уникнути ситуацій, коли покупці не можуть отримати бажаний товар через його відсутність, або коли підприємство залежить від непроданого запасу товару, що може призвести до фінансових втрат.

1.7. Висновки до першого розділу

У цьому розділі ми заглибилися у сферу програмного забезпечення для електронної комерції, розглянувши різні аспекти, які сприяють її успіху. Ми почали з огляду існуючих систем управління контентом (CMS), платформ управління взаємовідносинами з клієнтами (CRM) та аналітичних систем, які зазвичай використовуються в індустрії електронної комерції. Ці системи відіграють життєво важливу роль в управлінні контентом, взаємодією з клієнтами та отриманні інформації з даних.

Розглянуті поняття cross-sell і upsell є ефективними стратегіями маркетингу та продажу, спрямованими на збільшення прибутку компанії. Cross-sell полягає в пропозиції додаткових товарів або послуг під час основної покупки клієнта, тоді як upsell передбачає пропозицію більш дорогих або розширених версій продукту. Обидві стратегії стимулюють зростання середнього чеку, покращують задоволення клієнтів і сприяють загальному розвитку бізнесу. При правильному застосуванні і зверненні уваги на потреби клієнтів, cross-sell і upsell можуть бути потужними інструментами для підвищення продажів і витримки конкуренції на ринку.

Було досліджено підприємства, для яких розробка систем аналізу та прогнозування інформації була б дуже актуальною. Зокрема, підприємства електронної комерції отримують значну вигоду від використання цих систем, щоб отримати конкурентну перевагу на ринку. Використовуючи можливості аналізу та прогнозування даних, ці підприємства можуть приймати обґрунтовані рішення, оптимізувати свою діяльність та покращити досвід своїх клієнтів у здійсненні покупок.

Крім того, було розглянуто застосування методологій Data Science в індустрії електронної комерції. Методи Data Science, такі як інтелектуальний аналіз даних, машинне навчання та предиктивна аналітика, надають цінні інструменти для вилучення знань та прогнозування з величезних обсягів даних електронної комерції. Ці методології дозволяють компаніям виявляти закономірності, розуміти поведінку клієнтів і передбачати ринкові тенденції, що призводить до вдосконалення маркетингових стратегій, персоналізованих рекомендацій і збільшення доходів.

Очевидно, що інтеграція систем аналізу та прогнозування інформації на підприємствах електронної комерції має важливе значення для їхнього зростання та успіху. Впроваджуючи надійні CMS, CRM та аналітичні системи, компанії можуть ефективно управляти своїм контентом, взаємовідносинами з клієнтами та отримувати цінну інформацію з даних. Крім того, використання методологій Data Science дає компаніям можливість приймати рішення на основі даних і залишатися попереду у висококонкурентному середовищі електронної комерції.

Оскільки технології продовжують розвиватися, а дані стають все більш доступними, для підприємств електронної комерції вкрай важливо бути в курсі новітнього програмного забезпечення, систем і методологій. Використовуючи системи аналізу та прогнозування інформації, ці компанії можуть відкрити нові можливості, глибше зрозуміти своїх клієнтів і процвітати в динамічному світі електронної комерції.

РОЗДІЛ 2. АНАЛІЗ СПОСОБІВ ІНФОРМАЦІЙНОГО АНАЛІЗУ ТА ПРОГНОЗУВАННЯ

2.1. Огляд використаних технологій

Для проектів інформаційного аналізу та прогнозування використовуються різноманітні технології та інструменти:

1. Машинне навчання (Machine Learning): Ця технологія дає можливість комп'ютерним системам автоматично навчатися та покращувати свою продуктивність на основі даних без явного програмування. Методи машинного навчання, такі як навчання з учителем, ненавчання та підсилення, використовуються для аналізу даних та прогнозування.
2. Штучний інтелект (Artificial Intelligence): Штучний інтелект включає в себе комп'ютерні системи, які здатні розуміти, тлумачити та аналізувати дані, а також приймати рішення на основі цих даних. Використовуються методи, такі як обробка природної мови, комп'ютерне зорове сприйняття та інші, для розуміння та аналізу текстів, зображень та інших типів даних.
3. Статистичний аналіз: Статистичні методи використовуються для виявлення кореляцій, залежностей та закономірностей в даних. Вони дозволяють робити висновки, проводити гіпотези та робити прогнози на основі статистичних моделей.
4. Бази даних: Для зберігання, організації та доступу до даних використовуються системи управління базами даних (СУБД). Ці системи дозволяють ефективно зберігати та опрацьовувати великі обсяги даних для подальшого аналізу та прогнозування.
5. Обробка природної мови (Natural Language Processing, NLP): Ця технологія дозволяє комп'ютерам розуміти та обробляти людську мову. Вона використовується для аналізу текстових даних, включаючи структуровані та неструктуровані тексти, електронні листи, соціальні медіа тощо. Обробка природної мови може включати завдання, такі як

виявлення сутностей, класифікація тексту, аналіз тональності, машинний переклад та багато інших.

6. Великі дані (Big Data): В проектах аналізу та прогнозування часто зустрічаються великі обсяги даних, що вимагають спеціальних технологій для їх зберігання, обробки та аналізу. Технології Big Data, такі як системи розподіленого зберігання даних (Hadoop, Apache Spark) та інструменти для обробки потокових даних, дозволяють ефективно опрацьовувати великі обсяги даних та отримувати інсайти.
7. Візуалізація даних: Це важливий компонент проектів інформаційного аналізу та прогнозування. Візуалізація даних дозволяє графічно відображати результати аналізу, моделей та прогнозів для легкого сприйняття та зрозумілості. Інструменти візуалізації даних, такі як Tableau, Power BI, Python бібліотека Matplotlib, дозволяють створювати графіки, діаграми, інтерактивні панелі та інші візуальні елементи.

Залежно від конкретної задачі та доступних ресурсів, можуть бути використані інші інструменти та методи для досягнення конкретних цілей.

Для реалізації кваліфікаційної роботи магістра роботи, було використано технології та бібліотеки для Data Science досліджень: платформа Anaconda, мова програмування Python, бібліотеки Pandas, Numpy, Scikit-Learn для візуалізації даних було використано бібліотеки matplotlib та Seaborn. Для відображення всіх даних та реалізації коду було використано інтерактивну обчислювальну платформу Jupyter Notebook. Для ознайомлення з використаними технологіями в аналізі даних, будуть надані відповіді на питання: як, і для чого була використана та чи інша технологія, мова, бібіліотека.

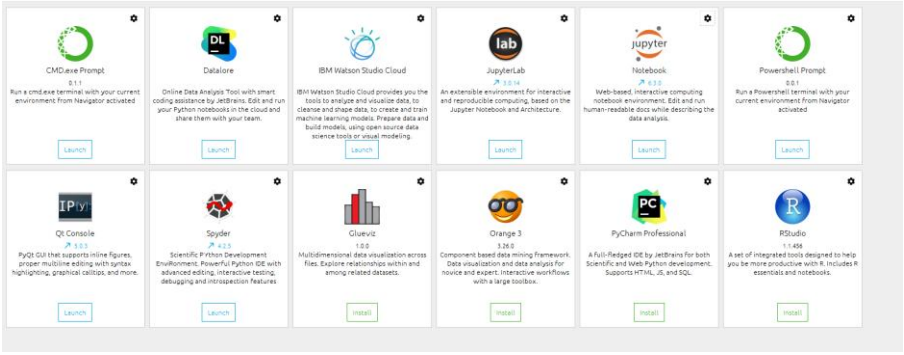
2.1.1. Мова програмування та платформа для реалізації проекту інформаційного аналізу та прогнозування

Anaconda - це комплексний, курируваний, високоякісний і високопродуктивний дистрибутив та менеджер пакетів для Python, R та багатьох супутніх пакетів для Linux, Windows та MacOS, призначений для використання вченими. За допомогою спеціалізованих середовищ Conda можна отримати великі колекції інструментів R та Python. Керування версіями та залежностями Conda забезпечує сумісність окремих компонентів у середовищі. Anaconda постачається з власним менеджером пакетів, який називається conda. Цей менеджер дозволяє легко встановлювати, оновлювати та керувати пакетами в даному середовищі. Anaconda поставляється з популярними бібліотеками для аналізу даних, такими як NumPy, Pandas, Matplotlib, SciPy, Scikit-learn та багатьма іншими, які забезпечують широкі можливості для обробки та аналізу даних.

Однією з особливостей Anaconda є можливість створення ізольованих середовищ, в яких можна встановлювати різні версії пакетів та бібліотек без взаємних конфліктів. Це дозволяє розробникам та дослідникам легко переключатись між різними проектами з різними залежностями.

Крім того, Anaconda надає користувачам доступ до інтегрованого розробницького середовища (IDE) Jupyter Notebook, що дозволяє створювати та виконувати інтерактивні документи, які поєднують код, текст та візуалізацію даних.

Загалом, Anaconda є потужним інструментом для наукових досліджень, аналізу даних та розробки програмного забезпечення, який допомагає спростити установку, керування пакетами та роботу з Python та іншими інструментами в наукових та аналітичних проектах. На даній платформі будувався весь проект аналізу даних електронної комерції. Реалізація коду



відбувалася в Jupyter Notebook – це інтерактивна обчислювальна платформа на основі веб-технологій.

Рисунок 2.1. Доступні стандартні модулі завдяки Anaconda

В розробці моделей, візуалізації, та роботі з наборами даних використовувалась мова програмування Python.

Python - інтерпретована, об'єктно-орієнтована, високорівнева мова програмування з динамічною семантикою, розроблена Гвідо ван Россумом. Вона вперше була представлена у 1991 році. Ця мова була розроблена як легкий і при цьому цікавий продукт. Назва «Python» є посиланням на британську комедійну групу Monty Python. Python має репутацію мови, зручної для початківців, замінюючи мову Java у ролі найбільш широко використовуваної вступної мови, оскільки більшу частину складнощів для користувача він бере на себе, дозволяючи новачкам зосередитися на повному розумінні концепцій програмування, а не на дрібних деталях. [4]

Які переваги має мова програмування Python та чому була обрана саме вона? Відповідь на це питання надає в своїй роботі Андреас Мюллер, він вказує, що Python поєднує в собі потужність мов програмування загального призначення з простотою використання специфічних мов, таких як MATLAB або R. У Python є бібліотеки для завантаження даних, візуалізації, статистики, обробки природної мови, обробки зображень та багато іншого. Цей великий інструментарій надає спеціалістам роботи з даними великий набір функцій загального та спеціального призначення. Однією з головних переваг використання Python є можливість прямої взаємодії з кодом, використовуючи термінал або інші інструменти, такі як JupyterNotebook, який ми розглянемо найближчим часом. Машинне навчання та аналіз даних за своєю сутністю є ітеративними процесами, в яких дані визначають перебіг аналізу. Для цих процесів дуже важливо мати інструменти, які дозволяють швидко виконувати ітерації та легко взаємодіяти з кодом. [20]

Python є мовою програмування загального призначення, що означає, що вона підходить для різноманітних задач. Вона використовується для веб-

розробки, наукових обчислень, аналізу даних, штучного інтелекту, машинного навчання, автоматизації задач, розробки ігор та багатьох інших областей.

Python також має велику та активну спільноту розробників, яка надає безліч сторонніх бібліотек та інструментів, що розширюють можливості мови. Наприклад, NumPy, Pandas, Matplotlib, TensorFlow, scikit-learn - це лише деякі з відомих бібліотек, що допомагають в аналізі даних та машинному навчанні.

Узагальнюючи, Python - це проста, потужна та широко використовувана мова програмування, яка підходить для багатьох завдань і має велику спільноту розробників, яка активно сприяє її розвитку та екосистемі.

2.1.2. Бібліотека Python для роботи з масивами NumPy

NumPy (Numerical Python) – це бібліотека для обробки масивів загального призначення. Вона надає високопродуктивний об'єкт багатовимірного масиву та інструменти для роботи з цими масивами. Вона є однією з основних бібліотек для наукових обчислень у Python і використовується широкою спільнотою дослідників, аналітиків даних та розробників.

Це основний пакет для наукових обчислень на Python. Він містить різні функції, включаючи такі важливі:

- потужний об'єкт N-мірного масиву;
- складні (широкомовні) функції;
- корисні можливості лінійної алгебри, перетворення Фур'є та випадкових чисел;
- крім очевидних наукових застосувань, NumPy можна також використовувати як ефективний багатовимірний контейнер загальних даних;
- довільні типи даних можна визначити за допомогою Numpy, що дозволяє NumPy легко і швидко інтегруватися з широким спектром баз даних.

Завдяки NumPy можна виконувати різноманітні математичні операції на масивах, такі як векторні обчислення, лінійну алгебру, трансформації Фур'є, статистичний аналіз, генерування випадкових чисел та багато іншого. Бібліотека також надає функції для інтеграції з C/C++ та Fortran кодом, що робить її потужним інструментом для високошвидкісних обчислень.

Окрім того, NumPy є основою для багатьох інших популярних бібліотек наукових обчислень в Python, таких як Pandas, SciPy, Matplotlib та інших. Вона інтегрується з ними, дозволяючи зручно використовувати їх функціонал разом з NumPy.

Загалом, NumPy є потужним інструментом для обробки та аналізу числових даних у Python. Вона надає широкі можливості для роботи з масивами та математичними операціями, що допомагає розробникам та дослідникам виконувати складні наукові обчислення з високою ефективністю. (Numerical Python) – це бібліотека для обробки масивів загального призначення. Вона надає високопродуктивний об'єкт багатовимірного масиву та інструменти для роботи з цими масивами. Вона є однією з основних бібліотек для наукових обчислень у Python і використовується широкою спільнотою дослідників, аналітиків даних та розробників.

2.1.3. Бібліотека pandas для аналізу даних

pandas (усі малі літери) - це популярний набір інструментів для аналізу даних мовою Python. У ньому представлений широкий спектр утиліт, починаючи від аналізу файлів різних форматів і закінчуючи перетворенням всієї таблиці даних в матричний масив NumPy. Це робить pandas надійним союзником у галузі науки про дані та машинного навчання.

Як і NumPy, pandas працює в основному з даними в одновимірних та двовимірних масивах; але pandas обробляє їх по-різному.

У pandas одновимірні масиви називаються серіями. Серія створюється за допомогою конструктора *pd.Series*, який має багато необов'язкових аргументів. Найпоширенішим аргументом є *data*, що задає елементи серії.

Pandas також надає інструменти для роботи з часовими рядами, включаючи функції для індексації та ресемплінгу даних за часом. Це робить її корисною для аналізу фінансових даних, датасетів з датами та інших даних, що залежать від часу.

Бібліотека Pandas також має вбудовану підтримку для обробки пропущених значень у даних, що дозволяє ефективно вирішувати проблеми з неповними або відсутніми даними. Вона надає зручний інтерфейс для виконання операцій з даними, таких як об'єднання, злиття, групування та агрегація.

Pandas інтегрується з іншими популярними бібліотеками для наукових обчислень, такими як NumPy, Matplotlib, SciPy, що робить її частиною потужного екосистеми наукових інструментів у Python.

Загалом, Pandas є незамінним інструментом для обробки, аналізу та маніпулювання даними у Python. Вона надає зручний інтерфейс для роботи з табличними даними, що дозволяє розробникам та аналітикам швидко і ефективно виконувати різноманітні завдання обробки даних.

Подібно до масивів NumPy, серії pandas також використовують ключове слово *dtype* для ручного приведення.

Pandas спрощує виконання багатьох трудомістких та повторюваних завдань, пов'язаних з роботою з даними, включаючи:

- очищення даних;
- заповнення даних;
- нормалізація даних;
- злиття та об'єднання;
- візуалізація даних;
- статистичний аналіз;
- перевірка даних;

```
anime = pd.read_csv('anime-recommendations-database/anime.csv')
```

	anime_id	name	genre	type	episodes	rating	members
0	32281	Kimi no Na wa.	Drama, Romance, School, Supernatural	Movie	1	9.37	200630
1	5114	Fullmetal Alchemist: Brotherhood	Action, Adventure, Drama, Fantasy, Magic, Mili...	TV	64	9.26	793665
2	28977	Gintama°	Action, Comedy, Historical, Parody, Samurai, S...	TV	51	9.25	114262
3	9253	Steins;Gate	Sci-Fi, Thriller	TV	24	9.17	673572
4	9969	Gintama'	Action, Comedy, Historical, Parody, Samurai, S...	TV	51	9.16	151266

– завантаження та збереження даних;

Рисунок 2.2. Приклад читання набору даних в форматі CSV завдяки pandas

2.1.4. Бібліотеки для візуалізації даних matplotlib та Seaborn

Matplotlib – це комплексна бібліотека для створення статичних, анімованих та інтерактивних візуалізацій мовою Python. Matplotlib робить легкі речі простими, а важкі – можливими.

Matplotlib дозволяє контролювати кожну аспект візуалізації, від налаштування осей координат до вибору кольорів, шрифтів та стилів ліній. Вона надає гнучкий інтерфейс для створення графіків з точною настройкою, що дозволяє створювати професійно виглядаючі візуалізації.

Matplotlib може бути використана для створення графіків як у статичному режимі, так і у взаємодіючому режимі, що дозволяє маніпулювати графіками та даними у реальному часі. Вона інтегрується з іншими бібліотеками для наукових обчислень, такими як NumPy та Pandas, що дозволяє зручно використовувати її функціонал у поєднанні з іншими інструментами аналізу даних.

Бібліотека Matplotlib дозволяє створювати графіки у різних форматах, включаючи зображення для вбудовування у документи, веб-сторінки або презентації, а також інтерактивні графічні візуалізації для використання у середовищі Jupyter Notebook або веб-додатках.

Завдяки Matplotlib можна:

– створюти графіки видавничої якості;

- створти інтерактивні фігури, які можна масштабувати, панорамувати, оновлювати;
- налаштовувати візуальний стиль та макет;
- експортувати до багатьох форматів файлів;
- вбудовувати в JupyterLab та графічні інтерфейси користувача;
- використовувати багатий набір пакетів сторонніх розробників, які побудовані на базі Matplotlib;

В цілому, Matplotlib є потужним інструментом для візуалізації даних у Python, який дозволяє створювати різноманітні та високоякісні графіки зі значною гнучкістю та контролем.

Seaborn – це бібліотека для створення статистичної графіки в Python. Вона побудована на базі matplotlib і тісно інтегрована із структурами даних pandas.

Seaborn допомагає досліджувати та розуміти дані. Його функції побудови графіків працюють з датафреймами та масивами, що містять цілі набори даних, і всередині виконують необхідне семантичне відображення та статистичне агрегування для створення інформативних графіків. Його орієнтований на набір даних декларативний API дозволяє зосередитись на тому, що означають різні елементи ваших графіків, а не на деталях того, як їх малювати.

Одним з ключових переваг Seaborn є його здатність до автоматичного визначення та стилізації графіків, що дозволяє швидко створювати графіки зі стандартними та привабливими оформленнями. Вона надає набір готових тем оформлення, які можна легко застосувати до графіків, що допомагає створювати єдиноформний та професійний вигляд візуалізацій.

Seaborn також має багато вбудованих функцій для створення різних типів графіків, таких як лінійні графіки, гістограми, діаграми розсіювання, ящики з вусами та інші. Вона дозволяє зручно візуалізувати взаємозв'язки між даними, розподіл значень, залежності та інші характеристики даних.

Крім того, Seaborn надає інструменти для статистичної візуалізації, таких як візуалізація розподілу даних, кореляційних матриць, регресійних моделей та

інших статистичних аналізів. Це робить бібліотеку корисною для проведення досліджень та аналізу даних з використанням візуальних методів.

Seaborn також інтегрується з іншими бібліотеками для наукових обчислень, такими як NumPy та Pandas, що дозволяє зручно працювати з даними та використовувати їх у візуалізаціях.

```
# Import seaborn
import seaborn as sns

# Apply the default theme
sns.set_theme()

# Load an example dataset
tips = sns.load_dataset("tips")

# Create a visualization
sns.relplot(
    data=tips,
    x="total_bill", y="tip", col="time",
    hue="smoker", style="smoker", size="size",
)
```

Рисунок 2.3. Приклад візуалізації даних за допомогою Seaborn

2.1.5. Бібліотека pylab для аналізу даних

PyLab - це модуль у мові програмування Python, який надає функціонал для виконання наукових обчислень та візуалізації даних. Він включає в себе багато інструментів і пакетів, таких як NumPy, Matplotlib та SciPy, і забезпечує зручний інтерфейс для роботи з цими пакетами.

Він дозволяє створювати графіки, діаграми, зображення та інші візуалізації даних. Він надає можливість швидкого відображення даних у вигляді графіків без необхідності додаткового коду, що спрощує процес аналізу та візуалізації даних.

Ця бібліотека використовує Matplotlib для побудови графіків і NumPy для роботи з масивами даних. Він також має деякі функції з пакету SciPy, які дозволяють виконувати наукові обчислення, такі як чисельна оптимізація, інтерполяція, розв'язування диференціальних рівнянь тощо.

PyLab є потужним інструментом для наукового обчислення і візуалізації даних у мові програмування Python. Він дозволяє швидко та зручно виконувати різноманітні аналітичні завдання і дослідження з використанням наукових пакетів.

2.1.6. Бібліотека Scikit-Learn для машинного навчання

Scikit-learn – це бібліотека аналізу даних з відкритим вихідним кодом та золотий стандарт машинного навчання (ML) в екосистемі Python. Ключові концепції та особливості включають:

Алгоритмічні методи прийняття рішень, включаючи:

- Класифікація: виявлення та категоризація даних на основі закономірностей.
- Регресія: передбачення або прогнозування значень даних на основі середнього значення існуючих та запланованих даних.
- Кластеризація: автоматичне угруповання схожих даних у набори даних.

Scikit-learn зосереджений на простоті використання, якості реалізації алгоритмів та дотриманні зрозумілого інтерфейсу. Вона надає розширену підтримку для навчання з учителем та навчання без учителя, включаючи популярні алгоритми, такі як Random Forest, Support Vector Machines, k-Nearest Neighbors, Naive Bayes, K-Means та багато інших.

Бібліотека Scikit-learn також надає функції для попередньої обробки даних, які допомагають у вирішенні проблем зі стандартизацією, нормалізацією, роботою з пропущеними значеннями та іншими типовими завданнями підготовки даних перед застосуванням моделей машинного

навчання, також пропонує інструменти для оцінки та порівняння моделей, включаючи метрики якості, розподіл даних для перехресної перевірки та підбір оптимальних гіперпараметрів моделей. Це допомагає вибрати найкращу модель та налаштувати її для досягнення найкращої продуктивності.

Дана бібліотека інтегрується з іншими популярними бібліотеками для наукових обчислень, такими як NumPy та Pandas, що дозволяє легко використовувати їх разом для аналізу та моделювання даних.

Алгоритми, що підтримують прогностичний аналіз, варіюються від простої лінійної регресії до нейромережного розпізнавання образів. [5]

2.1.7. Середовище розробки Jupyter-Notebook

Jupyter-Notebook розширює консольний підхід до інтерактивних обчислень у якісно новому напрямку, надаючи веб-додаток, який підходить для відображення всього процесу обчислень: розробки, документування та виконання коду, а також передачі результатів. Блокнот Jupyter поєднує два компоненти:

Jupyter-Notebook - інструмент на основі браузера для інтерактивного створення документів, що поєднують пояснювальний текст, математику, обчислення та їхній багатий мультимедійний результат.

Документи блокнота - представлення всього вмісту, який можна побачити у веб-застосунку, включаючи входи та виходи обчислень, пояснювальний текст, математику, зображення та багаті медіа-представлення об'єктів. [6]

Один з головних елементів Jupyter Notebook - це клітинки, які можуть містити код, текст або графіки. Користувач може виконувати окремі клітинки коду, що дозволяє ітеративно виконувати та перевіряти результати. Результати виконання коду відображаються безпосередньо під клітинкою коду, що дозволяє зручно аналізувати та візуалізувати дані.

Jupyter Notebook підтримує багато мов програмування, але основною мовою, яка використовується, є Python. Він надає доступ до багатьох

популярних бібліотек для аналізу даних та наукових обчислень, таких як NumPy, Pandas, Matplotlib та Scikit-learn.

Основні можливості Jupyter-Notebook:

- редагування коду в браузері з автоматичним підсвічуванням синтаксису, відступами та завершенням/інтроспекцією вкладок;
- можливість виконання коду з браузера, при цьому результати обчислень прикріплюються до коду, що згенерував їх;
- відображення результатів обчислень за допомогою багатих медіа-уявлень, таких як HTML, LaTeX, PNG, SVG тощо. Наприклад, малюнки видавничої якості, що відображаються бібліотекою matplotlib, можуть бути включені до рядка;
- редагування в браузері насиченого тексту за допомогою мови розмітки Markdown, яка може надавати коментарі коду, не обмежується звичайним текстом;
- можливість легко включати математичні позначення в комірки розмітки за допомогою LaTeX, які відображаються за допомогою MathJax.

2.2 Огляд методів кластеризації для розробки моделі машинного навчання

Для вирішення завдань кластеризації на даний момент доволі популярними є алгоритми, що розбивають дані на так звані «кластери» (групи), які є близькими за значеннями. Усередині кожної групи повинні виявитися «схожі» об'єкти, а об'єкти різних групи мають бути якомога відміннішими. Головна відмінність кластеризації від класифікації у тому, що перелік груп чітко не заданий та визначається у процесі роботи алгоритму. [7]

Використання кластерного аналізу можна поділити на кілька етапів:

1. Вибір набору об'єктів для кластеризації.

2. Визначення безлічі змінних, якими будуть оцінюватися об'єкти у вибірці. За потреби – нормалізація значень змінних.
3. Обчислення значень міри схожості між об'єктами.
4. Застосування методу кластерного аналізу створення груп подібних об'єктів (кластерів).
5. Подання результатів аналізу.

Більшість алгоритмів кластеризації передбачають порівняння об'єктів між собою на основі певної міри близькості (подібності). Мірою близькості називається величина, що має межу і зростає зі збільшенням близькості об'єктів. Заходи подібності «винаходяться» за спеціальними правилами, а вибір конкретних заходів залежить від завдання, і навіть від шкали вимірів. Як міра близькості для числових атрибутів дуже часто використовується евклідова відстань, що обчислюється за формулою:

$$D(x,y) = \sqrt{\sum_i (x-y)^2} \quad (1)$$

Для категорійних атрибутів поширена міра подібності Чекановського-Серенсена та Жаккара:

$$(|\ t_1 \cap t_2\ |\ /\ |\ t_1 \cup t_2\ |) \quad (2)$$

Машинне навчання (ML) – це технологія, яка дозволяє комп'ютерам навчатися на основі вхідних даних та будувати/навчати прогностичну модель без явного програмування. ML є підмножиною штучного інтелекту (AI). [8]

Щоб комп'ютери могли навчатися без явного програмування, необхідні алгоритми. Алгоритми - це набори правил, що застосовуються до обчислень.

Основні поняття алгоритмів ML:

Представлення - це спосіб конфігурування даних таким чином, щоб їх можна було оцінити. Приклади включають дерева рішень, набори правил, екземпляри, графічні моделі, нейронні мережі, машини векторів підтримки, ансамблі моделей та інші.

Оцінка – за наявності гіпотези оцінка – це спосіб оцінити її достовірність. Приклади: точність, прогноз та відгук, квадрат помилки, ймовірність, апостеріорна ймовірність, вартість, маржа, ентропія k-L дивергенція та інші.

Оптимізація - процес налаштування гіперпараметрів з метою мінімізації помилок моделі за допомогою таких методів, як комбінаторна оптимізація, оптимізація з обмеженнями та ін. [9]

В якості ML-алгоритму можна використати метод k-середніх, слід розглянути набір переваг та недоліків цього методу - серед переваг:

- Відносно простий у реалізації.
- Масштабується на великі масиви даних.
- Гарантує збіжність.
- Легко адаптується до нових прикладів даних.
- Узагальнюється на кластери різних форм та розмірів, наприклад, еліптичні кластери.

Перечислимо й недоліки цього методу:

- Кількість кластерів необхідно вибирати власноруч.
- У методу присутня проблема з кластеризуванням даних різних розмірів та щільності.
- Масштабується за кількістю вимірів – є менш продуктивним коли між значеннями велика різниця.

Також можна розглянути інший метод k-найближчих сусідів (kNN). Основна ідея методу KNN полягає в тому, що схожі об'єкти повинні мати близькі відстані один від одного, тому кластери формуються шляхом групування об'єктів, що знаходяться близько один до одного. Вибір параметра k є важливою частиною методу, оскільки він впливає на формування кластерів та розділення даних. Для вибору кількості сусідів зазвичай використовується підхід емпіричної оцінки, де вибирається оптимальне значення k на основі аналізу даних та експериментів.

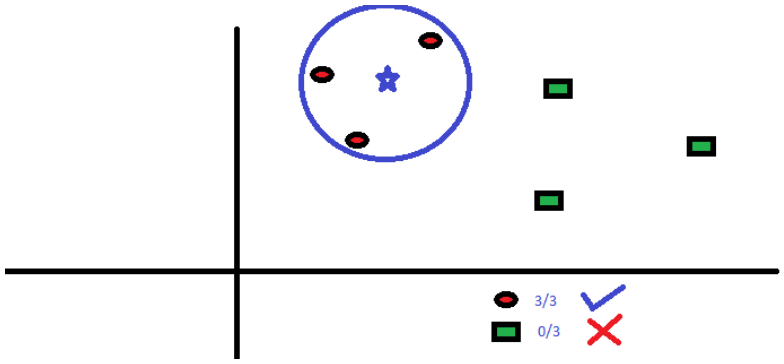


Рисунок 2.4. Приклад візуалізації роботи методу KNN

Метод KNN має кілька переваг, таких як простота реалізації, здатність працювати з даними різної форми та розміру, а також здатність добре працювати з шумом та вихідними даними невизначених розподілів. Однак, він також має певні обмеження, такі як чутливість до значень параметру k та обчислювальну складність для великих наборів даних.

У практичних застосуваннях метод KNN використовується для задач кластеризації, розпізнавання образів, рекомендаційних систем, класифікації та інших завдань машинного навчання, де потрібно групувати схожі об'єкти разом. Для того щоб порівняти обидва методи, слід перечислити переваги та недоліки й цього методу теж – серед переваг:

1. Простота kNN, ймовірно, є найпростішим алгоритмом машинного навчання і, можливо, найпростішим для розуміння.
2. Він є непараметричний - непараметричний метод означає, що kNN не робить припущень щодо набору даних. Якщо ви спочатку не знаєте занадто багато про набір даних, ця функція може бути рятівною.
3. Якщо ви хочете досліджувати ознаки зі складними зв'язками або якщо у ваших даних є викиди, які ви б хотіли врахувати, kNN може чудово впоратися з цим завданням.
4. Якщо ви хочете досліджувати ознаки зі складними зв'язками або якщо у ваших даних є викиди, які ви б хотіли врахувати, kNN може чудово впоратися з цим завданням.

Серед недоліків:

1. Дорогі обчислення - На жаль, kNN є голодним алгоритмом машинного навчання, оскільки йому доводиться обчислювати близькість між сусідами для кожного окремого значення набору даних.
2. Поглинає величезну кількість оперативної пам’яті - kNN зберігає всі свої значення в оперативній пам'яті, і знову ж таки, ви можете не помітити цього при невеликих реалізаціях, але спробуйте попрацювати з великою базою даних, і ви зрозумієте, що це не так.
3. Тільки невеликі розміри - якщо ви бажаєте працювати з наборами даних з великою кількістю ознак, це може бути проблематично з kNN.
4. Робота з відсутніми значеннями - kNN не може обробляти дані з пропущеними значеннями

Алгоритм Ball Tree є одним з методів, що можуть бути використані для ефективної реалізації методу k-найближчих сусідів (KNN). Він базується на ідеї створення структури даних, яка розділяє простір об'єктів на близькі групи, що спрощує пошук найближчих сусідів для даного об'єкта.

Побудова Ball Tree:

- Створення кореневого вузла дерева, який містить усі об'єкти набору даних.
- Рекурсивний процес побудови дерева шляхом розділення об'єктів на близькі групи (групи куль), які називаються вузлами дерева.
- Розділення об'єктів в кожному вузлі на дві підгрупи за допомогою медіанної точки та радіуса кулі, що найкраще розділяють об'єкти.
- Продовження рекурсивного побудови дерева для кожної підгрупи, поки не будуть виконані певні умови зупинки, наприклад, досягнення максимальної глибини дерева або мінімальної кількості об'єктів в кожному вузлі.

Пошук k найближчих сусідів:

- Знаходження початкового вузла дерева, який найближчий до цільового об'єкта.
- Рекурсивний процес спуску по дереву, перевіряючи вузли та їх кулі, щоб знайти k найближчих сусідів цільового об'єкта.
- Обчислення відстані між цільовим об'єктом та кожним вузлом/кулею та оновлення набору найближчих сусідів, якщо знайдено об'єкти, які є ближчими до цільового об’єкта.

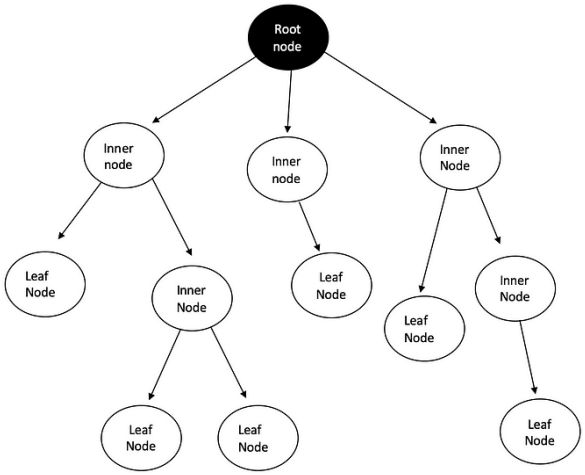


Рисунок 2.5. Структура алгоритму дерева Ball Tree методу KNN

Алгоритм Ball Tree дозволяє прискорити пошук найближчих сусідів у порівнянні зі стандартним пошуком за допомогою перебору, оскільки він зменшує кількість порівнянь об'єктів у просторі. Він особливо корисний для наборів даних з великою кількістю об'єктів і великою кількістю вимірів, де простий перебір стає обчислювально витратним.

Порівнюючи наведені два методи, можна дійти висновку, що куди більш значущі недоліки в роботі методу присутні у методу k-найближчих сусідів. Це і не дивно – в сьогоденні даний метод відходить на задній план і все рідше використовується на практиці. Це зумовлено тим, що дані які необхідно аналізувати збільшуються, а сам метод є неймовірно вимогливим до оперативної пам’яті та обчислювальних потужностях комп’ютеру, а також має проблему з роботою з великою кількістю даних. Провівши паралель між обома методами, можна дійти висновку, що метод k-середніх є більш доречним для

цілей завдання кластеризації покупців або поділу товарів за рейтингом, тому що ціллю розробки моделі є надання можливості швидко аналізувати як малі так і великі набори даних витрачаючи при цьому меншу кількість оперативної пам’яті та обчислювальних потужностей, а метод kNN, у свою чергу, може виконати простішу задачу у вигляді надання рекомендацій товарів, які стосуються вибору клієнта на основі схожості товарів, адже це займає менше обчислювальних потужностей.

Метод k-середніх(k-means) шукає заздалегідь певну кількість кластерів у непозначеному багатовимірному наборі даних. [10] Для цього використовується проста концепція оптимальної кластеризації:

"Центр кластера" - це середнє арифметичне всіх точок, що належать кластеру. [11]

Кожна точка знаходиться ближче до центру кластера, ніж до інших центрів кластера.

Ці два припущення є основою моделі k-means. Давайте поглянемо на простий набір даних і подивимося на результат k-means (рис 2.4).

```
from sklearn.datasets.samples_generator import make_blobs
X, y_true = make_blobs(n_samples=300, centers=4,
                       cluster_std=0.60, random_state=0)
plt.scatter(X[:, 0], X[:, 1], s=50);
```

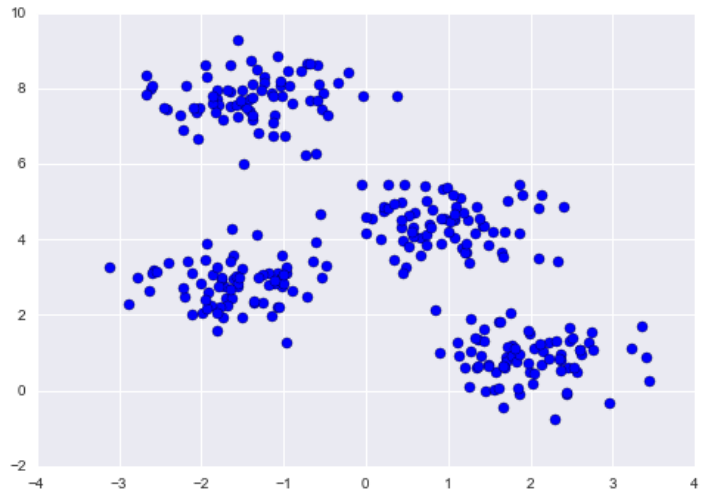


Рисунок 2.6. Результат k-середніх

Дія алгоритму така, що намагається мінімізувати сумарне квадратичне відхилення точок кластерів від центрів цих кластерів:

$$V = \sum_{i=1}^{k} \sum_{x \in S_i} (x - \mu_i)^2 \quad (3)$$

де k – число кластерів, Si – отриманні кластери, i = 1, 2, … , k, а μi – центри мас всіх векторів x із кластера Si.

Слід розглянути детальніше базові визначення та поняття алгоритму:

Метод k-середніх поділяє m спостережень на k груп (або кластерів) (k ≤ m) $\mathbf{S} = \{S_1, S_2, \ldots, S_k\}$ (4), щоб мінімізувати сумарне квадратичне відхилення точок кластерів від центроїдів цих кластерів:

$\min \left[\sum_{i=1}^{k} \sum_{x^{(j)} \in S_i} \left\| x^{(j)} - \mu_i \right\|^2\right]$ (5), де $x^{(j)} \in R^n, \mu_i \in R^n$ (6)

Кількість кластерів можна визначити на око, в даному випадку можна побачити явно 4 кластери, але визначення кількості кластерів «на око» не є оптимальним рішенням досвідченого аналітика. Для знаходження кількості кластерів можна скористатися методом різноманітними методами. [18]

Перший метод – це аналіз силуетів.

Оцінка силуету - це міра середньої подібності об'єктів у кластері та його відстані до інших об'єктів інших кластерах.

Для кожної точки даних i ми спочатку визначаємо:

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i, j) \quad (7)$$

які є середню відстань точки i до всіх інших точок, що належать до того ж кластеру Ci. Велике a(i) означає, що точка даних i несхожа свій кластер. Іншими словами, якщо точка i належить нульовому кластеру, це середня відстань точки i з усіма іншими точками нульового кластера.[19]

По-друге, ми визначаємо:

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i, j) \quad (8)$$

яке являє собою середню відстань точки і до решти точок наступного найближчого кластера. Велике b(i) означає, що точка даних і несхожа із сусіднім кластером.

Для цього необхідно виконати два кроки:

- Знайти всі середні відстані точки і до всіх інших точок, які не належать іншому кластеру (на відміну від Ci).
- Візьміть мінімальне із цих середніх значень.

Наприклад, якщо точка і належить нульовому кластеру, знайдіть середню відстань між точкою і та всіма іншими точками кластера 1, потім 2 тощо. Після обчислення всіх цих середніх відстаней береться мінімальна.

Нарешті, ми визначаємо силуетну оцінку точки даних і як:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \quad (9)$$

Глобальний показник силуету визначається як: $$S = \frac{1}{N} \sum_i s_i \quad (10)$$

Другий метод – це метод «ліктя».

Наступним методом, можна назвати метод ліктя, який і буде використано для методу кластеризації в цій магістерській роботі, завдяки простоті в реалізації, використовується для визначення оптимальної кількості кластерів. Як відомо, при збільшенні k-середнього спотворення зменшується, кожен кластер містить менше екземплярів, що входять до нього, та екземпляри розташовуються ближче до своїх центроїдів. Однак поліпшення середнього спотворення буде знижуватися зі збільшенням k. Значення k, у якому поліпшення спотворень знижується найсильніше, називається ліктем, у якому слід припинити розподіл даних на подальші кластери (рис 2.7). [12]

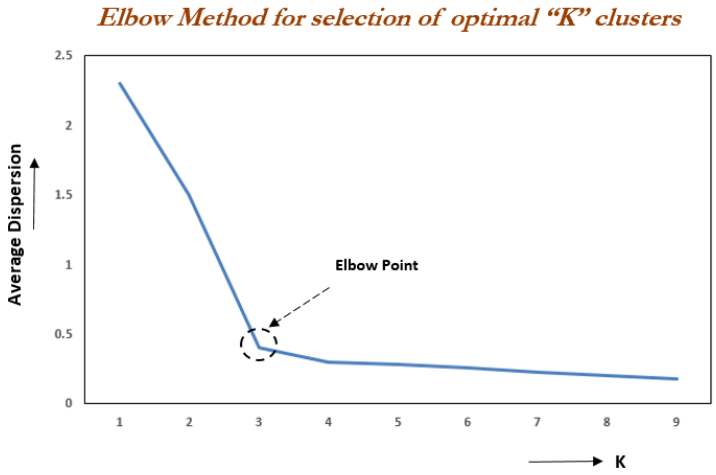


Рисунок 2.7. Використання методу «Ліктя» [13]

У методі «Ліктя» ми змінюємо кількість кластерів (K) від 1 до 10. Для кожного значення K ми розраховуємо WCSS (внутрішньокластерну суму квадратів). WCSS - це сума квадратів відстані між кожною точкою та центроїдом у кластері. Коли ми будуємо графік WCSS зі значенням K, графік виглядає як лікоть. У міру збільшення числа кластерів значення WCSS починає зменшуватись. Значення WCSS найбільше, коли K = 1. Аналізуючи графік, бачимо, що графік швидко змінюється у одній точці, створюючи цим форму ліктя. З цієї точки графік починає рухатися майже паралельно до осі X. Значення K, відповідне цій точці, є оптимальним значенням K або оптимальною кількістю кластерів. [12]

Давайте візуалізуємо результати, побудувавши графік даних, з пофарбованими відмітками. Також зобразимо центри кластерів, визначені за допомогою оцінювача k-means (рис 2.8):

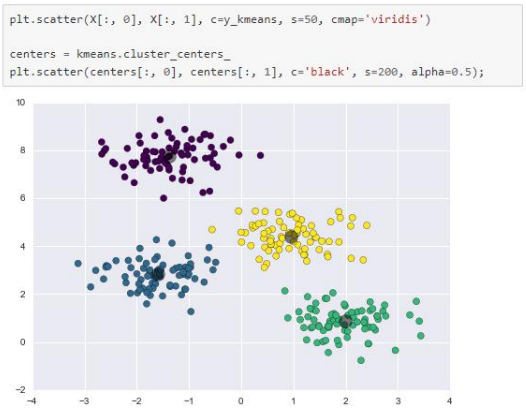


Рисунок 2.8. Візуалізація поділених кластерів [14]

Як ви можете бачити на рис 2.6, всього є 4 кластери, які візуалізуються різними кольорами, а центроїд кожного кластера візуалізується чорним кольором.

2.3. Огляд методу навчання асоціативним правилам для розробки моделі машинного навчання

Розглянемо також метод машинного навчання «Навчання асоціативних правил», що використовується для знаходження цікавих відношень між змінними у великих базах даних. Разом модель та інтелектуальний аналіз даних відносяться до галузі штучного інтелекту. Правило асоціації – це метод аналізу даних, типовим застосуванням якого є аналіз кошика покупок у супермаркеті. Основне завдання експертної системи – надання необхідної інформації працівнику, а завдання правила асоціації – виявлення цінного зв'язку між кожним елементом даних. Змінивши арифметику та метод складання правил, ми знаходимо вирішальне правило бази даних, яке може бути використане в експертній системі, і таким чином знаходимо метод пошуку вирішального правила за допомогою правил асоціації.

Навчання правилам асоціацій - це метод машинного навчання з урахуванням правил виявлення цікавих зв'язків між змінними у великих базах даних. Він призначений виявлення сильних правил, виявлених у базах даних, з допомогою деяких заходів цікавості. У будь-якій даній транзакції з безліччю елементів правила асоціації призначені виявлення правил, які визначають, як чи чому певні елементи пов'язані між собою.

Грунтуючись на концепції сильних правил, Ракеш Агравал, Томаш Імієліньскій і Арун Свамі[21] представили правила асоціації для виявлення закономірностей між продуктами у великомасштабних даних про транзакції, що реєструються системами точок продажу (POS) у супермаркетах. Наприклад,

правило $\{\text{onions}, \text{potatoes}\} \Rightarrow \{\text{burger}\}$ (10), знайдене в даних про продажі в супермаркеті, вказує на те, що якщо покупець купує цибулю та картоплю разом, то він, швидше за все, також купить м'ясо для гамбургера. Така інформація може бути використана як основа для прийняття рішень про маркетингові заходи, такі як, наприклад, рекламне ціноутворення або розміщення товару.

Навчання асоціативним правилам можна поділити на три типи алгоритмів:

- Apriori
- Eclat
- Алгоритм зростання F-P

Навчання асоціативним правилам працює на основі концепції

затвердження "Якщо" та "Інакше", наприклад, якщо А, то Б.

Рисунок 2.9. Приклад правила якщо – то.

Тут елемент If називається антецедента, а твердження Then – послідовником. Такі типи відносин, коли ми можемо виявити деяку асоціацію чи зв'язок між двома елементами, відомі як поодинока кардинальність. Вся справа у створенні правил, і якщо кількість елементів збільшується, то кардинальність також збільшується відповідно. Тому для виміру асоціацій між тисячами елементів даних існує кілька метриків. Ці метрики наведені нижче:

- Підтримка
- Довіра
- Ліфт

Підтримка - це частота або частота появи елемента в наборі даних. Вона визначається як частка транзакцій T, що містять набір елементів X. Якщо є X наборів даних, то для транзакцій T вона може бути записана як:

$$\text{Supp(X)}=\frac{Freq(X)}{T} \quad (11)$$

Довіра показує, як часто правило виявляється вірним. Або часто елементи X і Y зустрічаються разом у наборі даних, коли поява X вже дано. Це

$$\text{Confidence}=\frac{Freq(X,Y)}{Freq(X)}$$

відношення числа транзакцій, що містять X і Y, до записів, що містять X.

(12)

Ліфт це сила будь-якого правила, може бути визначена формулою:

$$\text{Lift}=\frac{Supp(X,Y)}{Supp(X)\times Supp(Y)} \quad (13)$$

Це відношення міри спостережуваної підтримки та очікуваної підтримки, якщо X та Y незалежні один від одного. Він має три можливі значення:

Якщо Lift=1: ймовірність появи антецедента та послідовника не залежить один від одного.

Lift>1: Визначає рівень залежності двох наборів елементів один від одного.

Lift<1: говорить нам про те, що один елемент є замінником інших елементів, що означає, що один елемент негативно впливає на інший.

Серед трьох доступних типів алгоритмів для Навчання правилам асоціації слід розглянути використаний в роботі – алгоритм росту F-P. Алгоритм зростання F-P розшифровується як Frequent Pattern, і це покращена версія алгоритму Apriori. Він є базою даних у вигляді деревоподібної структури, яка

відома як частий шаблон або дерево. Метою цього дерева частих шаблонів є вилучення шаблонів, що найчастіше зустрічаються.

Fp Growth – це модель Data Mining, заснована на правилах асоціації.

Ця модель дозволяє на основі історії транзакцій визначити набір асоціативних правил, що найчастіше зустрічаються, в наборі даних. Для цього як вхідний параметр їй необхідний набір транзакцій, що складається з кошиків продуктів, які клієнти вже придбали.

Враховуючи набір транзакцій, першим кроком FP-growth є обчислення частот елементів і визначення елементів, що часто зустрічаються.

На другому етапі FP-growth використовується структура суфіксного дерева (FP-дерева) для кодування транзакцій без явної генерації наборів-кандидатів, що зазвичай потребує великих витрат. Після другого кроку набори частих елементів можуть бути вилучені з дерева FP, і модель повертає набір правил асоціації продуктів, як показано в прикладі нижче:

```
{Product A + Product B} --> {Product C} with 60% probability
{Product B + Product C} --> {Product A + Product D} with 78% probability
{Prodcut C} --> {Product B + Product D} with 67% probability etc.
```

Щоб створити таблицю ймовірностей, моделі необхідно надати 2 гіперпараметри:

minSupRatio : мінімальна підтримка, щоб набір елементів був визначений як частий. Наприклад, якщо елемент зустрічається 3 рази із 5 транзакцій, його підтримка становить 3/5=0,6.

minConf: мінімальна довіра для створення асоціативного правила. Довіра - це показник того, як часто правило асоціації виявляється вірним. Наприклад, якщо в наборі транзакцій елементи X зустрічаються 4 рази, а X і Y зустрічаються лише 2 рази, то довіра до правила X => Y дорівнюватиме 2/4 = 0,5. Цей параметр не впливає на пошук частих наборів елементів, але визначає мінімальну довіру для генерації асоціативних правил на основі частих наборів елементів.

Після обчислення асоціативних правил залишається лише застосувати їх до кошиків товарів клієнтів.

2.4. Висновки до другого розділу

Для вирішення завдань аналізу даних та побудови моделей було проаналізовано існуючі інструменти, мови програмування та методи, які надають можливість аналітику даних швидко та зручно отримати доступ до необхідних йому даних. Завдяки мови програмування Python та платформи Anaconda можна отримати швидкий доступ до всіх необхідних інструментів та бібліотек для роботи в сфері Data Science. При виборі інструментів бралися до уваги напрацювання Андреаса Мюллера, в яких він описував, чому саме Python так добре підходить для виконання завдань Machine Learning.

Слід зазначити самі бібліотеки більш детально - завдяки matplotlib та seaborn можна візуалізувати дані, що цікавлять дослідника, а бібліотека sckitlearn надає доступ до швидкої реалізації методів машинного навчання, завдяки ж numpy з’являється можливість роботи з багатовимірними масивами даних.

В процесі аналізу наявних методів для вирішення задач науки про дані та надання необхідних даних для роботи моделі, були розглянуті найпопулярніші методи кластеризації з використанням Machine Learning такі як k-середніх та k-найближчих сусідів, завдяки всебічному аналізу переваг та недоліків кожного сформувався висновок, що для вирішення задачі моделювання більш за все підходить метод k-середніх, також було розглянуто методи для пошуку оптимальної кількості кластерів при застосуванні методу к-середніх, а саме: метод «силуету» та метод «ліктя», який і був обраний для реалізації в даній кваліфікаційній роботі магістра завдяки свої простоті в реалізації.

Також було розглянуто метод Навчання асоціативних правил з використанням алгоритму F-P Growth. За допомогою цього методу ідентифікуються сильні правила, які виявляються в базах даних з

використанням деяких вимірів зацікавленості. З використанням цього методу можна надавати імовірнісну інформацію для працівників та/або покупців з рекомендаціями, щодо наступних покупок/пропозицій.

РОЗДІЛ 3. ЗАСТОСУВАННЯ МЕТОДІВ АНАЛІЗУ ДАНИХ ДЛЯ РОЗРОБКИ МОДЕЛЕЙ ІНФОРМАЦІЙНОГО АНАЛІЗУ ТА ПРОГНОЗУВАННЯ

3.1. Аналіз даних у сфері електронної комерції

Провівши ретельний літературний огляд, було прийнято рішення, що для інформаційного аналізу та прогнозування будуть використані методи кластеризації k-середніх, k-найближчих сусідів та навчання правилам асоціації. Для пошуку кількості кластерів використовується метод «ліктя» тому що він є простішим в реалізації та добре працює, коли йде мова про невеличку кількість кластерів. В реалізації даних методів використовується мова Python з необхідними та наведеними в розділі 2 бібліотеками, які необхідні для роботи в сфері Data Science та Machine Learning. В якості оточення для виконання коду Python використовується JupyterNotebook.

Попередній аналіз даних є важливим етапом в аналітиці даних і відіграє ключову роль у підготовці даних для подальшого аналізу. Він допомагає зрозуміти характеристики даних, їх структуру, типи даних, розподіли, пропуски, аномалії та інші особливості. Попередній аналіз також включає перевірку якості даних, щоб виявити та виправити проблеми, такі як відсутні дані, дублікати, неправильні значення, викиди тощо. Це дозволяє забезпечити якість даних перед подальшим аналізом і уникнути спотворення результатів.

Окрім того, попередній аналіз даних допомагає визначити цілі аналізу та вибрати відповідні методи та моделі для виконання завдання. Він дозволяє виявити важливі залежності, патерни та характеристики даних, які можуть бути використані в подальшому аналізі. Крім того, попередній аналіз даних включає підготовку даних для аналізу шляхом очищення, перетворення, видалення несуттєвих атрибутів, розбиття на відповідні структури даних тощо. Це допомагає забезпечити оптимальність та придатність даних для виконання аналітики.

Для початкового розуміння даних підприємства електронної комерції, варто провести аналіз доступних даних. Для цього скористаємося даними товарів з відкритих джерел на прикладі книг. Для початку було проаналізовано дані популярності, рейтинг, відгуків та авторів книг, це необхідно для майбутньої організації платформи e-commerce:

Топ-10 книг з найбільшою кількістю відгуків:

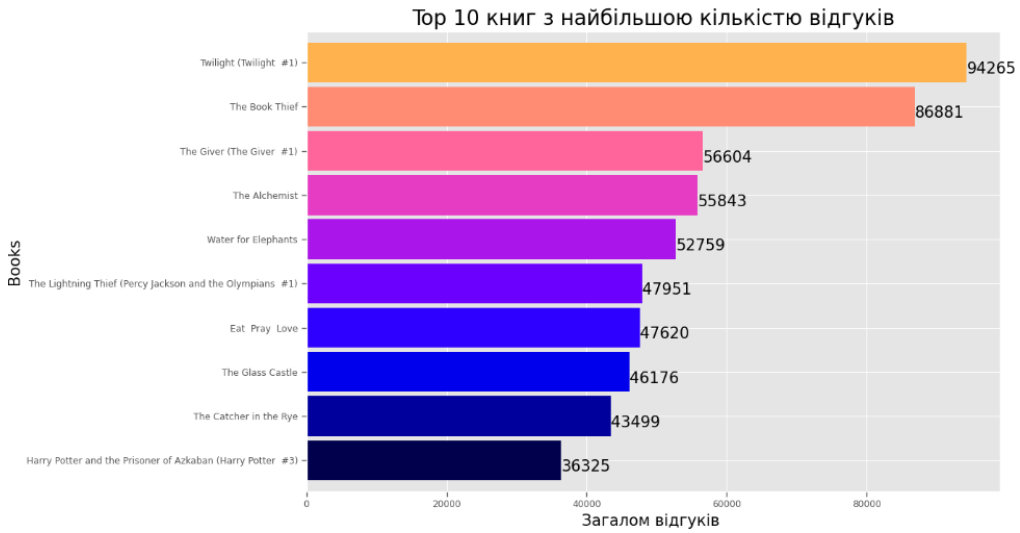


Рисунок 3.1. Гістограма за кількістю відгуків

Топ-10 книг з найбільшим рейтингом:

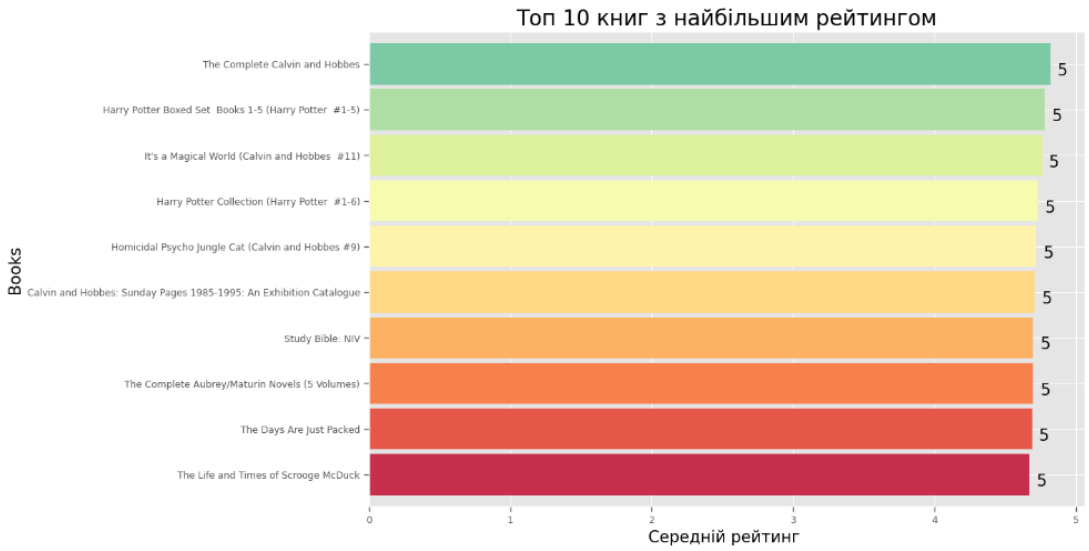


Рисунок 3.2. Гістограма за найбільшим рейтингом

Топ-10 письменників:

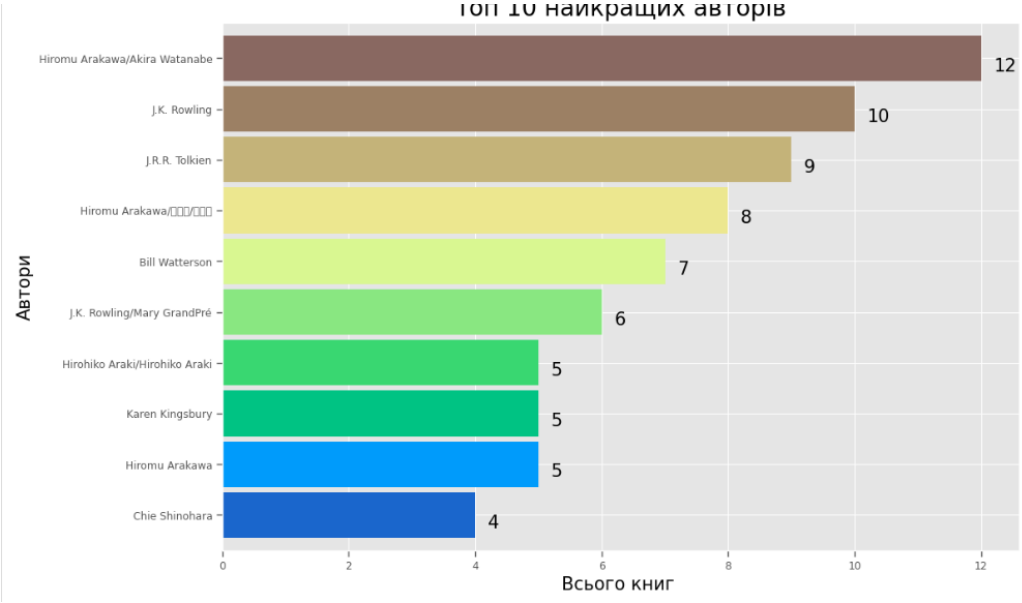


Рисунок 3.3. Гістограма найкращими авторами

3.2. Побудова моделей з використанням методу K-середніх

Для побудови першої моделі використовується набір даних з покупцями мережі електронної торгівлі. Спочатку слід розглянути, які колонки в наборі даних існують (рис. 3.14)

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40
5	6	Female	22	17	76
6	7	Female	35	18	6
7	8	Female	23	18	94
8	9	Male	64	19	3
9	10	Female	30	19	72

Рисунок 3.4. Таблиця з даними покупців

Дані взято з відкритого сховища kaggle. Серед даних, які надає датасет можна виділити такі основні колонки: Ідентифікатор покупця(*CustomerID*), Стать (*Gender*), Вік (*Age*), Щорічний прибуток(*Annual Income (k$)*), Оцінка витратоспроможності(*Spending Score*). Релевантними в даному випадку є

колонка прибутку покупця та оцінка його витратоспроможності. Ці колонки дадуть можливість визначити цільову аудиторію, яка може зацікавити маркетологів. Виділимо колонки, що нас цікавлять:

```
Ввод [5]: #Візьмемо колонки що нас цікавлять, а саме прибуток покупця та оцінку його витрат.
          X = dataset.iloc[:, [3,4]].values
```

Рисунок 3.5. Виділення необхідних колонок

Підключимо бібліотеку sklearn, а конкретно дістанемо функцію для роботи з методом кластеризації k-середніх, задамо очікувану максимальну кластерів в циклі, для пошуку серед значень необхідної нам кількості в майбутньому.

Для визначення необхідної кількості кластерів скористаємося методом «ліктя», для цього на відрізку з 10 значень знайдемо найменшу суму квадратів відстані між точками для якомога меншої кількості кластерів (рис. 3.16).

```
#Візуалізуємо "Метод ліктя" для того, щоб дізнатися, яка кількість кластерів буде оптимальною
plt.plot(range(1,11), wcss)
plt.title('Метод ліктя')
plt.xlabel('кількість кластерів')
plt.ylabel('сумма квадратів відстані між точками')
plt.show()
```

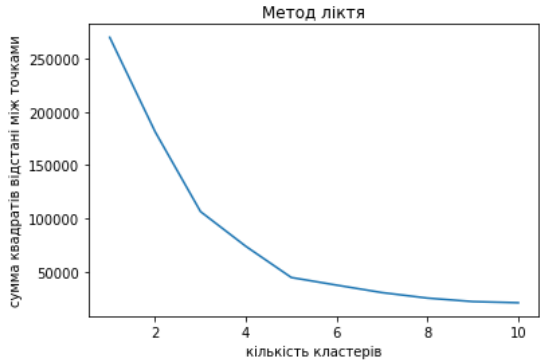


Рисунок 3.6. Графік методу «ліктя»

З графіку можна побачити, що найближчого необхідного нам значення крива досягає на значенні кластерів = 5. Знаходимо центроїди та розбиваємо точки на кластери за допомогою наступного коду:

```
kmeansmodel = KMeans(n_clusters= 5, init='k-means++', random_state=0)
y_kmeans= kmeansmodel.fit_predict(X)
```

Рисунок 3.17. Застосування методу k-середніх.

Після того як кількість кластерів визначена та робота алгоритму закінчена наступним кроком варто візуалізувати ці дані. Для цього скористаємося бібліотекою matplotlib, та призначимо кожному кластеру свій колір, для того, щоб отримані кластери можна було зручніше розрізнити. Для центроїдів задамо жовтий колір.

```
: #Візуалізуємо всі кластери

plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'red', label = 'Cluster 1')
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'blue', label = 'Cluster 2')
plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'green', label = 'Cluster 3')
plt.scatter(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], s = 100, c = 'cyan', label = 'Cluster 4')
plt.scatter(X[y_kmeans == 4, 0], X[y_kmeans == 4, 1], s = 100, c = 'magenta', label = 'Cluster 5')
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s = 300, c = 'yellow', label
plt.title('Кластери клієнтів')
plt.xlabel('Дохід (k$)')
plt.ylabel('Витрати')
plt.legend()
plt.show()
```

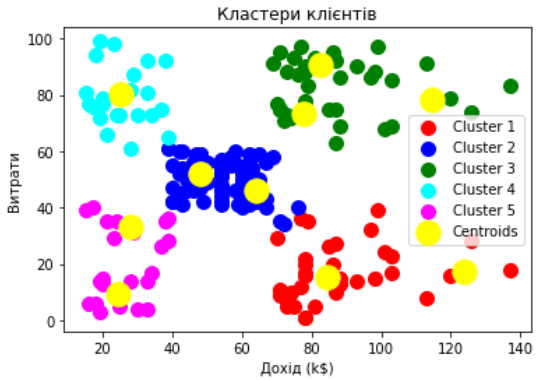


Рисунок 3.7. Виклик та результат роботи методу k-середніх.

Пояснення моделі:

– cluster 1 (Червоний колір) - високий прибуток, але мало тратять
– cluster 2 (Синій колір) - середній прибуток та витрати
– cluster 3 (Зелений колір) - високий прибуток та високі витрати. Є цільовою аудиторією. Для таких клієнтів є доцільним зробити рекламну розсилку по пошті та/або іншими засобами зв'язку
– cluster 4 (Голубий колір) - мало заробляють, але багато тратять
– cluster 5 (Рожевий колір) - мало заробляють, та мало витрачають

Останнім етапом знайдемо ідентифікатори яких покупців належать до якого кластеру, для того щоб пізніше їх можна було знайти в базі даних.

	CustomerID	cluster
0	1	4
1	2	3
2	3	4
3	4	3
4	5	4
5	6	3
6	7	4
7	8	3
8	9	4
9	10	3

```
cluster_map = pd.DataFrame()
cluster_map['CustomerID'] = dataset["CustomerID"]
cluster_map['cluster'] = y_kmeans
```

Рисунок 3.8. Виклик та результат роботи методу k-середніх.

Для побудови другої моделі використовується набір даних з рейтингом товарів мережі електронної торгівлі. Спочатку слід розглянути, які колонки в наборі даних існують:

	title	authors	average_rating	language_code	num_pages	ratings_count	text_reviews_count	publication_date	publisher
0	Harry Potter and the Half-Blood Prince (Harry ...	J.K. Rowling/Mary GrandPré	4.57	eng	652	2095690	27591	9/16/2006	Scholastic Inc.
1	Harry Potter and the Order of the Phoenix (Har...	J.K. Rowling/Mary GrandPré	4.49	eng	870	2153167	29221	9/1/2004	Scholastic Inc.

Рисунок 3.9. Огляд колонку другого датасету.

Перед початком розробки моделі для кластеризації варто переконатися, що між рейтингом та кількістю оцінок є зв’язок. Для цього побудуємо графік зв’язку між середньою оцінкою та кількістю оцінок.

```
In [170]: sns.set_context('paper')
          ax = sns.jointplot(x="average_rating", y="ratings_count",
          ax.set_axis_labels("Average Rating", "Ratings Count")

Out[170]: <seaborn.axisgrid.JointGrid at 0x1c3256d2ef0>
```

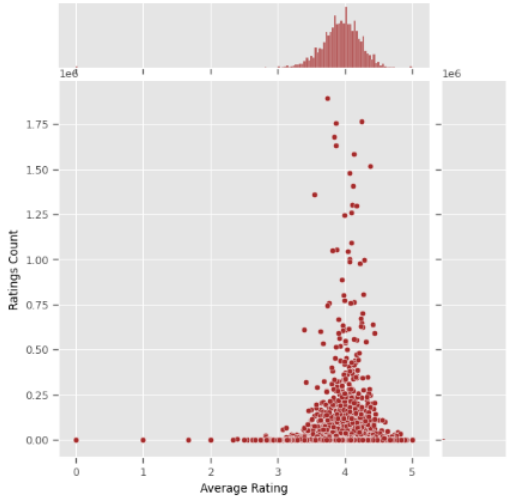


Рисунок 3.10. Графік зв’язку середнього рейтингу та кількості оцінок.

З графіка видно, що між середньою оцінкою та кількістю оцінок може існувати потенційний зв'язок. Зі збільшенням кількості оцінок рейтинг книги, здається, зменшується до 4. Середній рейтинг стає розрідженим, тоді як кількість оцінок продовжує зменшуватися.

Для того, щоб визначити кількість кластерів, яка необхідна для кластеризації k-середніх скористаємося знову методом «ліктя».

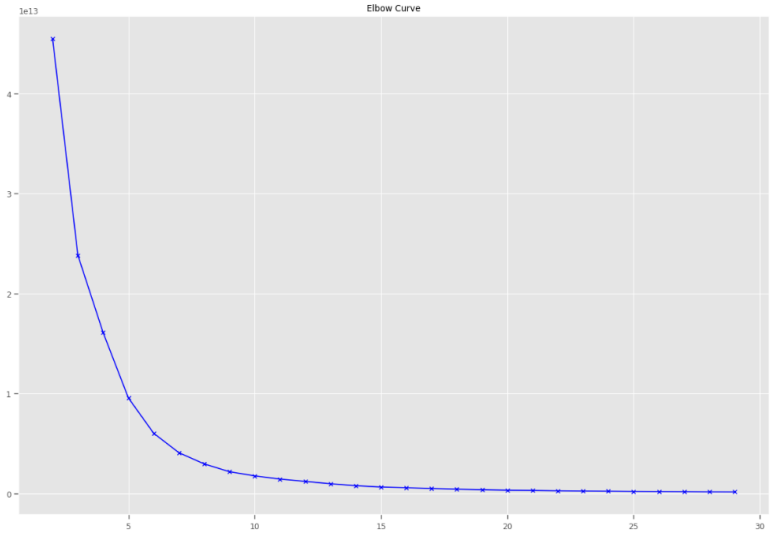


Рисунок 3.11. Графік «ліктя» для знаходження кластерів найпопулярніших товарів.

З наведеного вище графіка ми бачимо, що лікоть лежить навколо значення K=5, тому саме з цим значенням ми і спробуємо працювати. Обчислення K означає, що K = 5, таким чином, приймаючи його за 5 кластерів.

Після того як кількість кластерів визначена та робота алгоритму закінчена наступним кроком варто візуалізувати ці дані. Для цього скористаємося кожному кластеру свій колір, для того, щоб отримані кластери можна було зручніше розрізнити. Для центроїдів задамо зелений колір.

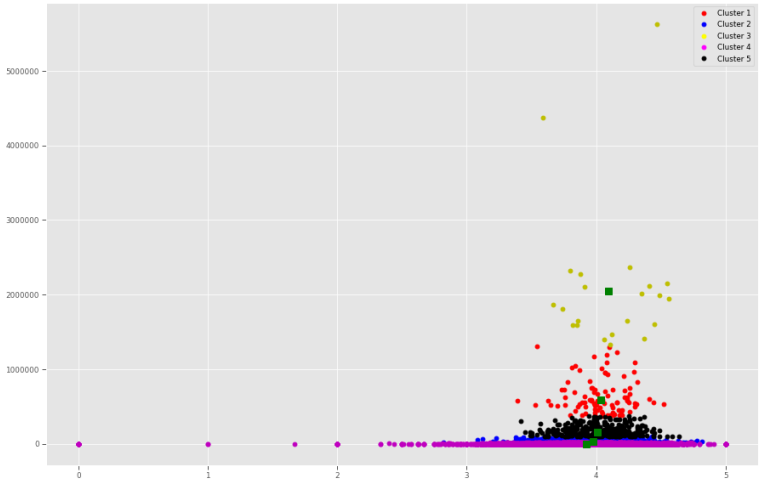


Рисунок 3.12. Робота методу кластеризації к-середніх з викидами.

З наведеного вище графіка видно, що через два викиди весь алгоритм кластеризації викривлений. Видалимо їх і сформуємо нові висновки. Викиди можна виявити за найбільшими значеннями.

```
# KMeans з оптимізацією

## Виявлення відхилень і їхнє усунення.

trial.idxmax()
```

```
Out[360]: average_rating     2034
          ratings_count     41865
          dtype: int64
```

```
In [361]:
trial.drop(41865, inplace = True)
```

Рисунок 3.11. Знаходження викидів.

Після того, як викиди очищені, можна запустити роботу методу ще раз, на цей раз без викидів, що можуть спортити результати.

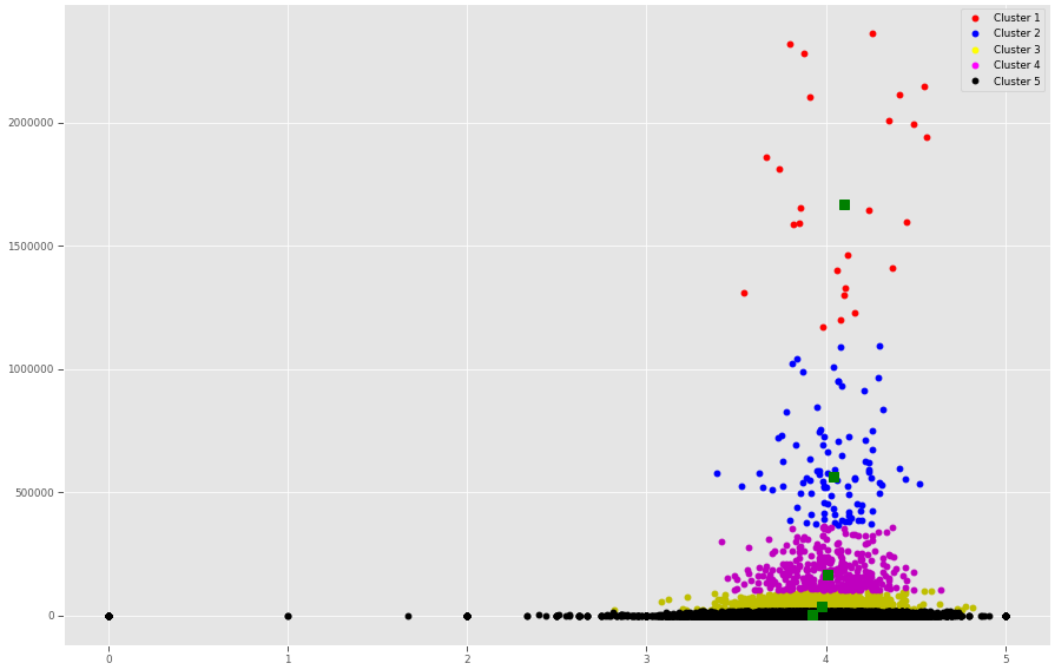


Рисунок 3.13. Результат кластеризації без викидів.

З наведеного вище графіка ми бачимо, що одразу всю систему можна класифікувати за кластерами. Зі збільшенням кількості балів рейтинг буде наближатися до наведеного вище кластеру. Зелені квадрати - це центроїди для даних кластерів. Зі зменшенням кількості рейтингів середній рейтинг стає більш розрідженим, з більшою волатильністю та меншою точністю.

Отримані дані є корисними, завдяки ним можна розділити загальні дані покупців на категорії різних груп за їх доходом та платоспроможністю на базі чого будувати маркетингову компанію, обирати стратегію розвитку бізнесу орієнтуючись на цільового покупця та використовувати для будь-яких інших цілей покращення роботи підприємства. Змоделюємо реальну ситуацію: на веб-сторінку магазину заходять два користувачі, умовно, покупець А – належить до кластеру рожевого кольору, які мало витрачають та мало заробляють, та покупець Б – з кластеру зелених точок, що заробляють багато та багато витрачають. Якщо запропонувати рекламне повідомлення користувачу А, яке буде рекламувати дорогий товар – користувач його скоріше за все не придбає, але якщо натомість запропонувати йому товар за незначною ціною та/або по знижці – шанс такої транзакції значно вищий. В ситуації з покупцем Б, уявімо,

що реклама так само як і покупцю А надається на дешевий товар, або товар зі знижкою – висока ймовірність, що він купить такий товар, але завдяки інформації, що цей покупець багато витрачає та заробляє, можна було б надати йому рекламу на гру популярного жанру, за стандартною ціною – тоді є висока ймовірність, що такий товар покупець Б теж придбає, а прибуток з продажу буде значно вищий.

Тепер, коли в нас є набір корисних даних, розробимо простий спосіб виводу даних, який зможе надати достатньо інформації для прийняття рішень. Для початку просумуємо вхідні точки та помітимо кожен кластер, який в нас

	cluster	wealth
1	81	середній прибуток та витрати
2	39	високий прибуток та високі витрати
0	35	високий прибуток, але мало тратять
4	23	малий заробіток, та малі витрати
3	22	малий заробіток, але високі витрати

утворився, у зручний табличний вигляд:

Рисунок 3.14. Табличний вигляд просумованих точок кластерів.

3.3. Побудова моделей з використанням методу K-найближчого сусіда

Побачивши кластеризацію, ми можемо зробити висновок, що можуть бути деякі рекомендації, які можуть відбуватися зі співвідношенням між середнім рейтингом та кількістю рейтингів.

Беручи Ratings_Distribution (Самостійно створений класифікаційний тренд), система рекомендацій працює з алгоритмом K найближчих сусідів.

На основі товару, введеного користувачем, найближчими сусідами до неї будуть класифіковані товари, які мають відношення до попереднього товару.

KNN використовується як для задач класифікації, так і для задач регресії. У задачах класифікації, щоб передбачити мітку екземпляра, ми спочатку

знаходимо k найближчих екземплярів до даного екземпляра на основі метрики відстані і на основі мажоритарного голосування або зваженого мажоритарного голосування (ближчі сусіди мають більшу вагу) ми передбачаємо мітки.

У такому середовищі відбувається навчання без нагляду, коли рекомендують схожих сусідів. У наведеному списку, якщо я попрошу порекомендувати "Над прірвою в житі", з'явиться п'ять книг, пов'язаних з нею.

Створення таблиці характеристик книг на основі Розподілу оцінок, яка класифікує, наприклад, книги за шкалою оцінок, як-от

- Між 0 та 1
- Від 1 до 2
- Від 2 до 3
- Від 3 до 4
- Від 4 до 5

Загалом, рекомендації враховують середні рейтинги та рейтинги за введеним запитом.

	Between 0 and 1	Between 1 and 2	Between 2 and 3	Between 3 and 4	Between 4 and 5	average_rating	ratings_count
bookID							
1	0	0	0	0	1	4.56	1944099
2	0	0	0	0	1	4.49	1996446
3	0	0	0	0	1	4.47	5629932
4	0	0	0	0	1	4.41	6267
5	0	0	0	0	1	4.55	2149872

Рисунок 3.15. Табличний вигляд рейтингу

Мін-максер використовується для зменшення зміщення, яке могло б виникнути через те, що деякі книги мають велику кількість ознак, а решта - меншу. Мін-макснер скалер знайде медіану для всіх книг і вирівняє її.

```
In [49]: min_max_scaler = MinMaxScaler()
         books_features = min_max_scaler.fit_transform(books_f
         eatures)
```

Рисунок 3.16. Виклик мін-макс скелера.

Створимо модель з використанням методу к-найближчого сусіда з кількістю сусідів – 6.

In [51]:
```
model = neighbors.NearestNeighbors(n_neighbors=6, alg
orithm='ball_tree')
model.fit(books_features)
distance, indices = model.kneighbors(books_features)
```

Рисунок 3.17. Створення моделі.

Створення спеціальних функцій для пошуку назв книг:

- Отримати індекс з назви
- Отримати ідентифікатор з часткової назви (оскільки не всі можуть запам'ятати всі назви)
- Показати схожі товари з набору даних ознак. (Для цього використовується метрика Індекси з найближчих сусідів для вибору товарів).

In [52]:
```
def get_index_from_name(name):
    return df[df["title"]==name].index.tolist()[0]

all_books_names = list(df.title.values)

def get_id_from_partial_name(partial):
    for name in all_books_names:
        if partial in name:
            print(name,all_books_names.index(name))

def print_similar_books(query=None,id=None):
    if id:
        for id in indices[id][1:]:
            print(df.iloc[id]["title"])
    if query:
        found_id = get_index_from_name(query)
        for id in indices[found_id][1:]:
            print(df.iloc[id]["title"])
```

Рисунок 3.18. Знаходження товарів за назвою.

3.4. Побудова моделі з використанням навчання асоціативних правил

Для побудови моделі машинного навчання асоціативних правил потрібно збирати вхідні дані, які містять інформацію про транзакції або події, для яких будуть встановлюватися асоціації. Ці дані можуть бути у вигляді таблиці або набору транзакцій.

Попередня обробка даних також є важливою складовою процесу побудови моделі асоціативних правил. Цей крок включає в себе очищення даних, видалення дублікатів, нормалізацію, видалення шуму та інші операції, які покращують якість та коректність даних.

Після попередньої обробки даних можна переходити до побудови моделі асоціативних правил. Це включає в себе застосування алгоритмів, таких як Apriori або FP-Growth, які знаходять часті набори або правила в даних. Ці алгоритми використовуються для виявлення зв'язків та асоціацій між різними елементами або подіями.

Для побудови моделі використано набір даних покупки в мережі електронної комерції, взяті з відкритого сховища kaggle. Розглянемо колонки та стрічки, які існують в даному датасеті.

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	GroupPrice
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850.0	United Kingdom	15.30
1	536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850.0	United Kingdom	20.34
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850.0	United Kingdom	22.00
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850.0	United Kingdom	20.34
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850.0	United Kingdom	20.34

Рисунок 3.19. Використаний набір даних для навчання асоціативним правилам.

Пояснення змінних:

- InvoiceNo: Номер рахунку, який відповідає покупці продукту.
- StockCode: Ідентифікатор придбаного продукту. Кожен ідентифікатор свій.
- Description: Опис придбаного продукту.
- Quantity: Кількість придбаного продукту.
- InvoiceDate : Дата рахунку, з 01/12/2010 до 09/12/2011.

- UnitPrice: Ціна одного продукту.
- CustomerID: Ідентифікатор покупця. Кожен ідентифікатор свій.
- Country: Країна, де клієнт розмістив замовлення.
- GroupPrice: Ціна всіх куплених однакових продуктів. Кількість х Ціна за одиницю

Далі необхідно провести попередню обробку даних, цей етап складається з двох кроків:

1. Видалення продуктів, що відповідають подарункам, що пропонуються компанією клієнтам. Ми зберігаємо тільки ті продукти, які клієнт справді поклав у свій кошик.
2. Ми групуємо всі продукти, придбані клієнтом разом. Кожен рядок відповідає транзакції, що складається з номера рахунку, ідентифікатора клієнта та всіх придбаних продуктів.

Після попередньої обробки даних, отримуємо наступний набір даних в

	basket	next_product	proba
199	{22920, 22917, 22919, 22921, 22916}	{22918}	0.992537
424	{22917, 22919, 22921, 22916}	{22918}	0.986014
306	{22917, 22918, 22921, 22920}	{22916}	0.985714
310	{22917, 22921, 22920, 22916}	{22918}	0.985714
96	{22917, 22919, 22921, 22920}	{22918}	0.985401
232	{22919, 22921, 22920, 22916}	{22918}	0.985401
197	{22918, 22920, 22917, 22919, 22921}	{22916}	0.985185
202	{22918, 22920, 22919, 22921, 22916}	{22917}	0.985185
365	{22917, 22921, 22916}	{22918}	0.979866
456	{22917, 22919, 22921}	{22918}	0.979730

якості результату:

Рисунок 3.20. Таблиця даних після попередньої обробки

Щоб створити цю таблицю, моделі необхідно надати 2 гіперпараметри: minSupRatio – мінімальна підтримка для визначенного набору елементів від частоти. minConf - мінімальна довіра для створення асоціативного правила.

```
a=time.time()
freqItemSet, rules = fpgrowth(basket['StockCode'].values, minSupRatio=0.005, minConf=0.3)
b=time.time()
# print('time to execute in seconds : ',b-a, ' s.')
# print('Number of rules generated : ', len(rules))

association=pd.DataFrame(rules,columns =['basket','next_product','proba'])
association=association.sort_values(by='proba',ascending=False)
print('Вимірність таблиці асоціацій : ', association.shape)
association.head(10)
```

```
Вимірність таблиці асоціацій :  (4958, 3)
```

	basket	next_product	proba
199	{22920, 22917, 22919, 22921, 22916}	{22918}	0.992537
424	{22917, 22919, 22921, 22916}	{22918}	0.986014
306	{22917, 22918, 22921, 22920}	{22916}	0.985714
310	{22917, 22921, 22920, 22916}	{22918}	0.985714
96	{22917, 22919, 22921, 22920}	{22918}	0.985401
232	{22919, 22921, 22920, 22916}	{22918}	0.985401
197	{22918, 22920, 22917, 22919, 22921}	{22916}	0.985185
202	{22918, 22920, 22919, 22921, 22916}	{22917}	0.985185
365	{22917, 22921, 22916}	{22918}	0.979866
456	{22917, 22919, 22921}	{22918}	0.979730

Рисунок 3.21. Згенеровані правила асоціації

Наступним кроком необхідно з корзини користувача повернути продукт для рекомендації, якщо він не був знайдений у списку асоціацій таблиці, пов'язаної з моделлю FP Growth. Для цього ми шукаємо в таблиці асоціацій продукт для рекомендації кожного окремого продукту в кошику споживача. Далі, для кожного клієнта в наборі даних ми шукаємо відповідну асоціацію у таблиці моделі Fp Growth. Якщо асоціацію не знайдено, ми викликаємо функцію compute_next_best_product, яка шукає асоціацію окремих продуктів.

Якщо індивідуальні асоціації не знайдено, функція повертає (0,0).

Порахуємо йомвірність наступної покупки для кожного клієнта:

```
Ввод [33]: a=time.time()
           list_next_pdt, list_proba= find_next_product(basket)
           b=time.time()
           print(b-a)
           basket['Recommended Product']=list_next_pdt # Набір рекомендованих продуктів
           basket['Probability']=list_proba # набір ймовірності асоціацій
           basket.head()
```

```
135.70527625083923
```

Out[33]:

InvoiceNo	CustomerID	StockCode	Recommended Product	Probability
536365	17850.0	[22752, 71053, 84029G, 21730, 85123A, 84406B, ...	0	0.000000
536366	17850.0	[22632, 22633]	22865	0.516393
536367	13047.0	[21755, 22748, 22623, 48187, 22745, 22310, 226...	22750	0.593516
536368	13047.0	[22912, 22960, 22913, 22914]	22961	0.322280
536369	13047.0	[21756]	21754	0.576132

Рисунок 3.22. Ймовірності покупок для клієнтів.

Результуючим набором даних є таблиця з рекомендаціями для кожного замовлення зробленого в мережі електронної комерції, доступ до яких можна отримати за ідентифікатором кошику, в результаті чого модель порекомендує товар, який відділу продажів варто зарекомендувати покупцю наступним:

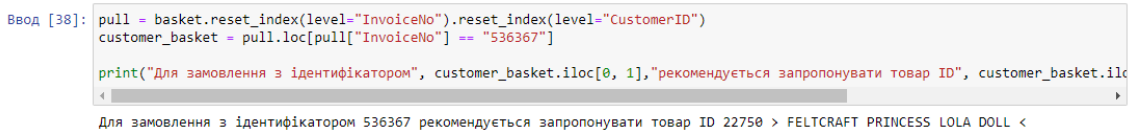


Рисунок 3.23. Результат роботи методу.

3.5. Висновки до третього розділу

В процесі аналізу даних було реалізовано метод k-середніх для поділу покупців на кластери за значеннями їх даних про витрати та щорічний заробіток та для поділу товарів за рейтингом. До цього, була реалізована система рекомендації схожих товарів з використанням методу k-найближчого сусіда.

Також було реалізовано метод навчання асоціативних правил, для знаходження знаходження цікавих відношень між змінними у базі даних з кошиками покупців.

Завдяки бібліотекам matplotlib та seaborn були візуалізувані дані, а завдяки бібліотеці skitlearn було реалізовано модель з використанням методу k-середіх, що виявилася доволі ефективною для поставлених цілей, не є вимогливою до ресурсів комп’ютеру порівняно з методом k-найближчих сусідів.

З результату роботи методу k-середніх було отримано 5 кластерів, що поділяють покупців на різні групи:

Кластер 1 -> високий прибуток, але мало тратять

Кластер 2 -> середній прибуток та витрати

Кластер 3 -> високий прибуток та високі витрати

Кластер 4 -> мало заробляють, але багато тратять

Кластер 5 -> мало заробляють, та мало витрачають

В реалізації моделі, ці групи покупців будуть використані для її побудови за допомогою застосування простих наборів правил, за результатом яких буде надана експертна оцінка.

З результату роботи методу навчання асоціативних правил було визначено рекомендації для покупців в залежності від замовлень, які вони здійснювали. Для кожного кошику з відповідним набором покупок було знайдено рекомендовану наступну покупку, з вказаною вірогідністю здійснення цієї покупки. Надалі, за допомогою ідентифікатору кошику, покупець чи працівник відділу продажів зможе дізнатися наступний рекомендований товар.

У цьому розділі ми детально розглянули принципи та алгоритми кожного з цих методів. Вони можуть бути використані окремо або в поєднанні з іншими методами для досягнення кращих результатів. Вибір конкретного методу залежить від задачі, типу даних та інших факторів. Засвоєння цих методів дозволяє нам розширити наші знання та навички в галузі машинного навчання та аналізу даних і створити ефективні моделі для розв'язання різноманітних завдань.

РОЗДІЛ 4. ОПИС РОБОТИ МОДЕЛЕЙ ІНФОРМАЦІЙНОГО АНАЛІЗУ І ПРОГНОЗУВАННЯ ТА РЕКОМЕНДАЦІЇ ЩОДО ЗАСТОСУВАННЯ

4.1. Опис роботи моделі з застосуванням методу k-середіх та методика застосування

За метою створення моделі, які відносяться до першого набору даних, це ті що добуває корисні знання за допомогою застосування методу к-середніх, є створеною для надання корисних даних власникам підприємств електронної комерції.

Метод кластеризації товарів за рейтингом та кількістю відгуків може бути корисним інструментом для впровадження стратегії cross-sell. Цей метод дозволяє групувати товари, що мають схожі рейтинги та кількість відгуків, у спільні кластери.

Застосовуючи цей метод, компанія може ідентифікувати товари, які часто придбаються разом або доповнюють один одного. Наприклад, якщо виявляється, що товари з високим рейтингом та великою кількістю відгуків часто придбаються разом, то компанія може використовувати цю інформацію для пропозиції додаткових товарів з цього кластеру під час продажу основного товару.

Цей підхід допомагає зрозуміти поведінку клієнтів і визначити, які товари можуть бути цікавими для них. Наприклад, якщо клієнт придбав товар з високим рейтингом та багатьма відгуками з певного кластеру, продавець може запропонувати йому інші товари з цього ж кластеру, які можуть бути йому цікаві.

Застосування методу кластеризації дозволяє збільшити ефективність стратегії cross-sell, оскільки компанія може точніше визначити, які товари мають великий потенціал для комбінування та додаткових продажів. В

результаті, це може підвищити середній чек клієнтів і забезпечити більше прибутку для компанії.

З набору даних, який повертає робота методу k-середніх можна згенерувати відповідь для кінцевого користувача:

Ввод [179]:
```
def generate_expert(cluster_amount):
    if cluster_amount.index[0] == 1:
        target = ["товари середньої вартості", "високої вартості"]
    if cluster_amount.index[0] == 0 or cluster_amount.index[0] == 2:
        target = ["дорогі товари", "низької вартості"]
    if cluster_amount.index[0] == 3 or cluster_amount.index[0] == 4:
        target = ["дешеві товари", "високої вартості"]
    return print('\n Цільовими покупцями для онлайн-магазину є покупці, в яких',

generate_expert(cluster_amount)
```

Рисунок 4.1. Зображення коду генерації відповіді.

Користуючись якою, в свою чергу, працівник отримує наступний результат з проаналізованих даних:

«Цільовими покупцями для онлайн-магазину є покупці, в яких середній прибуток та витрати ,тому орієнтуючись на даних покупців, є доцільним на титульній сторінці сайту опублікувати товари середньої вартості та розробити рекламні баннери орієнтуючись на покупців даного достатку.

Найменшою часткою покупців магазину є покупці, в яких малий заробіток, але високі витрати , тому товари високої вартості варто прибрати з титульної сторінки, або зменшити зайняте ними місце на екрані покупця.

Ці дії дозволять більш успішно продавати найбільш популярні серед користувачів товари для збільшення середнього прибутку підприємства електронної комерції.»

Модель, що використовує метод k-середніх для поділу клієнтів по їхньому прибутку та витратам, дозволяє класифікувати клієнтів на основі їх фінансових характеристик. Вона використовує дві важливі метри - прибуток, який вказує на дохід, отриманий від клієнта, та витрати, які показують, скільки клієнт витрачає на продукти або послуги.

Ця модель може мати широке застосування у сфері маркетингу та управління клієнтськими взаєминами.

Модель може допомогти в сегментації клієнтів на основі їхнього прибутку та витрат. Наприклад, можна створити кластери клієнтів з високим прибутком та високими витратами, що вказуватиме на важливих та прибуткових клієнтів, або кластери з низьким прибутком та високими витратами, що може вказувати на потенційно нерентабельних клієнтів.

Також, ця модель може допомогти розробити персоналізовані стратегії маркетингу та збуту для різних клієнтських сегментів. Наприклад, для клієнтів з високим прибутком та низькими витратами можна використовувати стратегії збуту високоякісних продуктів, тоді як для клієнтів з низьким прибутком та високими витратами можна пропонувати акційні пропозиції та знижки.

Вона може використовуватися для прогнозування прибутковості нових клієнтів на основі їх фінансових характеристик. Це допоможе компанії прийняти рішення про прийняття або відхилення нових клієнтів з урахуванням потенційної рентабельності.

Ця модель може бути корисною для аналізу фінансових даних клієнтів та прийняття рішень, спрямованих на оптимізацію відносин з клієнтами та збільшення прибутковості.

Побудуємо мережу виведення для моделі за допомогою зворотного виведення. Зворотний вивід починається з переліку цілей (або гіпотез) і працює в зворотному напрямку від висновку до антецеденту, щоб побачити, чи доступні дані, які будуть підтримувати будь-який з цих висновків.

Механізм логічного виводу, що використовує зворотний вивід, шукає серед правил виводу перше правило, у якого висновок (частина Тоді) відповідає поставленій меті. Якщо невідомо, чи набуває антецедент (частина Якщо) цього правила логічного значення "істина", тоді антецедент цього правила додається до списку цілей. (для того, щоб мета підтвердилася, необхідно також отримати дані для підтвердження цього нового правила).

На прикладі з набором даних про витрати та витратоспроможність покупців метою є знаходження необхідного логічного виводу з наявних даних.

1. Якщо покупців Х витратоспроможності, більше інших – тоді товари відповідної цінової категорії повинні бути виведенні на головну сторінку.
2. Якщо покупці Х витратоспроможності мають малий заробіток – тоді Х покупці можуть купити лише не велику кількість товарів, тому кількість дорогих позицій має бути зменшена.

Механізм логічного виводу, що використовує зворотний вивід, шукає серед правил виводу перше правило, у якого висновок (частина Тоді) відповідає поставленій меті. Якщо невідомо, чи набуває антецедент (частина Якщо) цього правила логічного значення "істина", тоді антецедент цього правила додається до списку цілей. (для того, щоб мета підтвердилася, необхідно також отримати дані для підтвердження цього нового правила).

Друга побудована модель з застосуванням методу k-середніх кластеризує товари за рейтингом та кількістю оцінок. Метод кластеризації k-середніх за рейтингом та кількістю оцінок товарів може допомогти в покращенні стратегій cross-sell та upsell шляхом ідентифікації спільних характеристик серед клієнтів та їхніх покупок.

Після застосування методу кластеризації k-середніх до даних про товари, можна сформувати кластери, де товари зі схожими рейтингами та кількістю оцінок будуть групуватись разом. Це дозволить нам отримати категорії товарів зі схожими характеристиками та оцінками.

Після цього, для стратегії cross-sell, можна виявити товари з високим рейтингом та багатьма оцінками і пропонувати їх як додаткові або супутні товари під час покупки клієнта. Наприклад, якщо клієнт придбав смартфон з високим рейтингом, система може рекомендувати йому чохол або навушники, які також мають високі оцінки.

У випадку з upsell, можна надавати пропозиції клієнтам про покупку більш високоякісних або дорожчих товарів, основуючись на їхньому рейтингу

та кількості оцінок. Наприклад, якщо клієнт придбав певний електронний пристрій з високим рейтингом, система може рекомендувати йому більш потужну або функціональну модель зі схожими характеристиками.

Отже, метод кластеризації k-середніх за рейтингом та кількістю оцінок товарів допомагає виокремити групи товарів зі схожими характеристиками, що дозволяє здійснювати більш персоналізовані та зорієнтовані на конкретних клієнтів.

4.2. Опис роботи моделі з застосуванням методу k-найближчих сусідів та методика застосування

Модель, що використовує метод k-середніх для поділу товарів по рейтингу та кількості оцінок, дозволяє групувати товари на основі їхньої популярності та якості. Вона аналізує рейтинг, який відображає оцінку користувачів, а також кількість оцінок, яка вказує на масштаб популярності товару серед користувачів.

Застосування моделі можуть бути різноманітними. Наприклад, вона може використовуватись у рекомендаційних системах для підбору товарів, які відповідають попереднім вподобанням користувачів. Модель може групувати схожі товари за їхнім рейтингом та кількістю оцінок, допомагаючи знаходити товари зі схожими характеристиками, що можуть зацікавити користувача.

Метод k-найближчих сусідів (k-nearest neighbors, KNN) для рекомендацій за рейтингом, кількістю оцінок та назвою може бути застосований для стратегії cross-sell, зокрема для пропозиції додаткових товарів або послуг клієнтам під час їх основної покупки.

Метод KNN базується на ідеї, що подібні об'єкти мають подібні властивості або характеристики. В контексті cross-sell, це означає, що якщо клієнт придбав певний товар існуючої категорії, можна використати KNN для ідентифікації і рекомендації інших товарів, які були придбані іншими клієнтами зі схожими вподобаннями. Цей підхід дозволяє пропонувати клієнту

товари, які ймовірно будуть йому цікаві, основуючись на вподобаннях та поведінці подібних клієнтів. Це сприяє зростанню продажів, стимулює cross-sell та поліпшує загальний досвід покупця.

Використання методу KNN для cross-sell допомагає компаніям аналізувати інформацію про покупки клієнтів та розуміти, які товари мають великий потенціал для комбінування та додаткових продажів. Це забезпечує персоналізований підхід до рекомендацій та може збільшити задоволення клієнтів та прибуток компанії.

Також, модель може використовуватись у маркетингових дослідженнях для аналізу ринку та конкуренції. Вона може допомогти виявити популярні товари з високим рейтингом та великою кількістю оцінок, що може свідчити про їхню успішність на ринку. Таким чином, компанії можуть зосередитись на розвитку та просуванні цих товарів, що сприятиме підвищенню їхньої конкурентоспроможності.

Застосування моделі з методом k-середніх для класифікації товарів за рейтингом та кількістю оцінок може бути корисним і в інших галузях, де необхідно групувати товари за їхніми характеристиками та популярністю.

Застосування моделі з методом k-середніх для поділу товарів по рейтингу та кількості оцінок може бути корисним для електронної комерції. Наприклад, онлайн-магазин може використовувати цю модель для розподілу свого товарного асортименту на групи залежно від популярності та якості товарів. Це допоможе покупцям знайти товари, які відповідають їхнім потребам та вимогам, а також підвищить ефективність продажів для магазину.

Ця модель може бути використана для аналізу ринку та конкуренції в певній галузі. Вона допоможе виявити та порівняти популярні товари у порівнянні з конкурентами на основі їхнього рейтингу та кількості оцінок. Це дасть змогу компаніям зорієнтуватися на ринку, підвищити свою конкурентоспроможність та вдосконалити свою стратегію маркетингу.

Крім вищезазначених застосувань, модель з методом k-середніх для поділу товарів по рейтингу та кількості оцінок може також бути використана у

соціальних мережах. Наприклад, платформи, які пропонують відео контент, можуть використовувати цю модель для рекомендацій відео на основі популярності та задоволення користувачів. Це допоможе покращити персоналізацію контенту та збільшити залучення аудиторії до платформи.

Загалом, модель з методом k-середніх для поділу товарів по рейтингу та кількості оцінок відкриває широкі можливості для аналізу та використання даних про товари. Вона допомагає зрозуміти поведінку користувачів, виявити популярні та якісні товари, покращити рекомендації та стратегії маркетингу, а також збільшити ефективність бізнесу.

4.3. Опис роботи моделі з застосуванням методу навчання асоціативних правил та методики застосування

Умова кожного правила визначає зразок деякої ситуації, при дотриманні якої правило може бути виконано. У даного методу є своя особливість – правилам вона навчається сама, та не потребує ручного введення правил фахівцем, за винятком введення двох гіперпараметрів:

- minSupRatio: мінімальна підтримка, щоб набір елементів був визначений як частий.
- minConf: мінімальна довіра для створення асоціативного правила.

Пошук рішення полягає у виконанні тих правил, зразки яких зіставляються з поточними даними.

За зв’язком з реальним часом модель є динамічною, тобто інтерпретує ситуацію, працює у поєднанні з датчиками об'єктів у режимі реального часу з постійною інтерпретацією даних, що надходять.

За вирішуваним завданням модель класифікується, як та, що виконує завдання Прогнозування. Прогнозуючі системи логічно виводять вірогідні наслідки з заданих ситуацій. У прогнозуючій системі зазвичай використовується параметрична динамічна модель, в якій значення параметрів

«підганяються» під задану ситуацію. Виводяться з цієї моделі, складають основу для прогнозів з ймовірними оцінками.

Система рекомендацій з використанням методу навчання асоціацій може бути ефективним інструментом для стратегій upsell та cross-sell. Метод асоціацій дозволяє виявляти кореляції та залежності між різними товарами або послугами, що часто придбуваються разом, і на цій основі робити рекомендації клієнтам.

Для upsell, система може аналізувати покупки клієнтів і виявляти групи товарів, які часто придбуваються разом або мають високу взаємодію. За допомогою цих асоціацій можна рекомендувати клієнтам більш висококласні або дорожчі версії товарів, які вони вже мають намір придбати. Наприклад, якщо клієнт купує ноутбук, система може рекомендувати йому модель з більшим обсягом пам'яті або потужнішим процесором.

У випадку cross-sell, система може розпізнавати схожі товари або послуги, які клієнти придбувають разом, і рекомендувати додаткові продукти з цього асоційованого або подібного кластеру. Наприклад, якщо клієнт купує смартфон, система може порекомендувати йому захисну плівку, чохол або навушники, які часто придбуваються разом зі смартфоном.

Застосування методу навчання асоціацій для системи рекомендацій дозволяє пропонувати клієнтам релевантні товари або послуги, які можуть бути їм цікаві. Це стимулює зростання середнього чеку та поліпшує здатність до cross-sell та upsell. Крім того, система може надавати персоналізовані рекомендації, що поліпшує загальний досвід покупця та задоволення клієнтів.

Враховуючи поведінку покупців та аналізуючи асоціації між товарами або послугами, система рекомендацій може стати потужним інструментом для досягнення цілей upsell та cross-sell, збільшення прибутку компанії та покращення взаємодії з клієнтами.

Для вирішення задач з використанням методу навчання асоціативних правил представити знання, які надає робота методу, можна продукційною

моделлю:

{Продукт А + Продукт В} --> {продукт С} з імовірністю 60%

{ Продукт В + Продукт t С} --> { Продукт А + Продукт D} з імовірністю 78%

{ Продукт С} --> { Продукт В + Продукт D} з імовірністю 67%.

Або ж якщо наводити більш практичний приклад: Якщо покупець купив {Картоплю + Цибулю} тоді покупець придбає {Олію} з імовірністю Х.

Правила для знаходження імовірностей генеруються за допомогою машинного навчання моделі на відповідному наборі даних про покупки за весь час, тому побудова механізму виведення перепадає на роботу самого метода, від фахівця необхідно лише введення порогового значення мінімальної затребуваності, що використовується для знаходження всіх частот предметів у базі даних та обмеження на мінімальну впевненість, що застосовується до частот наборів предметів для утворення правил. Але можна побудувати сітку частот наборів предметів, які купують.

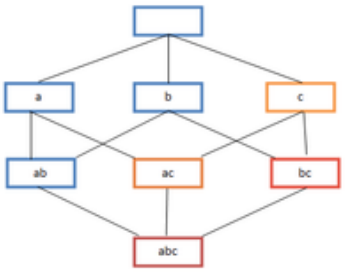


Рисунок 4.2. Сітка частот наборів предметів.

За допомогою розробленої моделі працівник відділу продажів може за введеним ідентифікатором кошику зарекомендувати покупцю наступний товар. Наприклад по запиту з ідентифікатором кошику «536367» модель надає наступну рекомендацію: «Для замовлення з ідентифікатором 536367 рекомендується запропонувати товар ID 22750 > FELTCRAFT PRINCESS LOLA DOLL < з імовірністю покупки 59%». (рис 4.27)

```
Ввод [39]: pull = basket.reset_index(level="InvoiceNo").reset_index(level="CustomerID")
           customer_basket = pull.loc[pull["InvoiceNo"] == "536367"]

           print("Для замовлення з ідентифікатором", customer_basket.iloc[0, 1],"рекомендується запропонувати това

           Для замовлення з ідентифікатором 536367 рекомендується запропонувати товар ID 22750 > FELTCRAFT PRINC
           ESS LOLA DOLL < з імовірністю 0.5935162094763092
```

Рисунок 4.3. Результат роботи моделі з використанням методу навчання асоціативним правилам.

Ця модель здатна аналізувати великі обсяги даних, зокрема історичні дані про покупки користувачів, з метою виявлення зв'язків і асоціацій між різними товарами. Вона використовує алгоритми машинного навчання, зокрема методи асоціаційних правил, для автоматичного виявлення цих залежностей.

Застосування такої моделі може бути різноманітним у сфері електронної комерції. Наприклад, вона може використовуватись для рекомендацій товарів покупцям на основі їхніх попередніх покупок та звичок. Це допомагає забезпечити персоналізовані рекомендації, що підвищують задоволення покупців та сприяють збільшенню продажів.

Модель може бути використана для крос-продажів, тобто пропозиції додаткових товарів, які часто купуються разом з основним товаром. Шляхом виявлення асоціацій між товарами, модель може запропонувати покупцям додаткові продукти, які доповнюють їхні покупки та задовольняють їхні потреби.

Ця модель може бути використана для персоналізованого маркетингу та створення цільових пропозицій для окремих сегментів покупців. Аналізуючи зв'язки між товарами та вподобанням користувачів, модель може рекомендувати спеціальні акційні товари, знижки або промо-коди, що залучають та стимулюють покупців до продовження покупок.

Таким чином, модель, що застосовує метод навчання асоціативних правил для рекомендації товарів у сфері електронної комерції, виявляється потужним інструментом для покращення персоналізації, збільшення продажів та підвищення задоволення покупців.

4.4. Висновки до четвертого розділу

В даному розділі було описано моделі, що були розроблені для надання рекомендацій користувачам та/або працівникам сфери електронної комерції.

Вони надають корисну інформацію, автоматизуючи роботу підприємства та замінюючи роботу експерта.

Модель з застосуванням методу k-середніх надає рекомендаційну інформацію для заповнення титульної сторінки інтернет-магазину виходячи з даних про кількість покупців з певною покупною спроможністю та заробітками кожного, поділених на кластери. Модель, що використовує метод k-середніх, виявилася цінним інструментом для задач кластеризації та сегментації. Ітеративно розподіляючи точки даних по кластерах на основі їхньої схожості з центроїдами кластерів, ця модель дає змогу виявляти значущі закономірності та групування в наборах даних. Вона знайшла застосування в різних сферах, включаючи сегментацію клієнтів, маркетингові дослідження та виявлення аномалій.

Модель, що використовує метод k-найближчих сусідів, є універсальним алгоритмом, який використовується для задач класифікації та регресії. Використовуючи поняття близькості, ця модель визначає клас або значення нової точки даних на основі міток або значень її найближчих сусідів. Вона широко використовується в системах рекомендацій, розпізнавання образів і виявлення відхилень, дозволяючи робити точні прогнози і приймати рішення на основі схожих прикладів у даних.

Моделі, що використовують метод навчання на основі асоціативних правил, особливо корисні для виявлення взаємозв'язків і асоціацій між елементами або змінними в наборах даних. Вивчаючи повторюваність і залежності між різними елементами, ці моделі отримують цінну інформацію і генерують дієві правила. У сфері електронної комерції вони успішно застосовуються для аналізу ринкових кошиків, персоналізованих рекомендацій та стратегій перехресних продажів.

Загалом, ці моделі відіграють вирішальну роль у завданнях аналізу інформації та прогнозування, дозволяючи компаніям і дослідникам видобувати значущу інформацію, робити точні прогнози та отримувати цінні висновки зі складних наборів даних. Розуміючи основні принципи та застосування цих

моделей, фахівці-практики можуть використовувати їхні можливості для прийняття рішень на основі даних, оптимізації процесів і підвищення загальної продуктивності в різних сферах.

ВИСНОВОК

В процесі підготовки до виконання роботи було сформульовано мету дослідження, об’єкт та предмет роботи та сформульовано основні етапи процесу виконання роботи, також були проаналізовані літературні джерела, які надали велику кількість корисної інформації в процесі.

В першому розділі магістерської роботи було розглянуто поняття електронної комерції загалом, були перелічені та описані засоби для моніторингу та збору статистики в сфері e-commerce, такі як: модулі CMS, CRM Google Analytics.

Були розглянуті приклади підприємств, для яких могло б знадобитися запровадження подібних систем та наведено приклади та причини необхідності їх використання.

В другому розділі були розглянуті основні інструменти для роботи в сфері Data Science з використанням мови Python. В якості платформи для проведення роботи було використано Anaconda – це платформа з великою кількістю дистрибутивів для Data Science проектів з використанням мов Python та R. Дана платформа забезпечила проект необхідними для аналізу та візуалізації бібліотеками для Python: Seaborn, matplotlib, numpy, pandas та бібліотеку для машинного навчання Scikit-learn. Було описано за допомогою чого були оброблені та візуалізовані дані, а саме за допомогою оточення для виконання коду Python – JupyterNotebook. В процесі аналізу наявних методів для вирішення задач науки про дані були розглянуті найпопулярніші методи кластеризації з використанням Machine Learning такі як k-середніх та k-найближчих сусідів, завдяки всебічному аналізу переваг та недоліків кожного сформувався висновок, що для вирішення першої задачі моделювання більш за все підходить метод k-середніх робота якого полягає в мінімізації сумарного квадратичного відхилення точок кластерів від центроїдів цих кластерів та метод навчання асоціативним правилам для вирішення другої задачі було використано метод навчання асоціативних правил, що використовується для

знаходження цікавих відношень між змінними у великих базах даних за допомогою сильних правил, які виявляються в базах даних з використанням деяких вимірів зацікавленості.

Третій розділ - це етап моделювання, який описує, як було реалізовано використані методи, наприклад, метод кластеризації k-середніх для пошуку цільових покупців судячи з даних їх прибутків та витрат, кількість кластерів при використані якого було визначено за допомогою методу «Ліктя». В результаті виконання роботи було досягнуто необхідних результатів роботи методу, а саме були виділені кластери точок, що відображали покупців на графіку, з якого були отримані дані для подальшого використання в розробці моделей. Також в третьому розділі описується реалізація методу навчання асоціативних правил для знаходження рекомендованого товару покупцеві в залежності від складу його кошику. Правило асоціації – це метод аналізу даних, типовим застосуванням якого є аналіз кошика покупок у супермаркеті. Основне завдання експертної системи – надання необхідної інформації працівнику, а завдання правила асоціації – виявлення цінного зв'язку між кожним елементом даних.

Ця робота має перспективи для розширення, адже моделі можуть бути використані для будь-якої кількості даних та в різних сферах бізнесу, наприклад знаходження найбільш ефективних працівників по кільком параметрам оцінки або знаходження найрозумніших учнів за їх оцінками та/чи іншими параметрами оцінювання знань.

Останній розділ надає описову інформацію щодо розроблених систем. Кожна система була класифікована за кожним типом, було формалізовано бази знань для них та наведені приклади роботи даних експертних систем з наданням рекомендаційної інформації для покупців/працівників сфери електронної комерції.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Information Technology and Implementation (Satellite): Conference Proceedings, December 01, 2022, Kyiv, Ukraine / Taras Shevchenko National University of Kyiv and [etc]; Vitaliy Snytyuk (Editor). – Kyiv: Publisher Individual entrepreneur Picha Y.V., 2022. 186 p.
2. Google Analytics Fundamentals. URL: https://medium.com/analytics-for-humans/what-is-google-analytics-and-why-is-it-important-to-my-business-8c083a9f81be (дата звернення: 03.05.2023).
3. Глорія Філіпс-ен. Advances in Data Science: Methodologies and Applications, 2020. С. 44-48.
4. Python - документація. URL: https://docs.python.org/3/ (дата звернення: 03.05.2023).
5. Партап Дангети. Statistics for Machine Learning, 2017. С. 74-83.
6. Уес Мак-Кінні. Python for Data Analysis, 2011. С. 33-35.
7. JupyterNotebook – документація. URL: https://jupyter-notebook.readthedocs.io/en/stable/ (дата звернення: 03.05.2023).
8. Джузеппе Бонаккорсо. Machine Learning Algorithms, 2017. С. 124-132.
9. Орельен Герон. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 2017. 17 с.
10. Адітья Бхаргава. Grokking algorithms, 2016. С. 21-25.
11. Чару Агарвал, Чандан Реді. Data Clustering – Algorithms and Applications, 2014. С. 325-365.
12. Цзюньцзе У. Advances in K-means Clustering: A Data Mining Thinking, 2012.
13. Python Data Science Handbook. URL: https://jakevdp.github.io/PythonDataScienceHandbook/05.11-k-means.html (дата звернення: 03.05.2023).
14. Statistics for Machine Learning handbook – The elbow method. URL: https://www.oreilly.com/library/view/statistics-for-

machine/9781788295758/c71ea970-0f3c-4973-8d3a-b09a7a6553c1.xhtml(дата звернення: 03.05.2023).

15. Факультет прикладних наук – дослідження ринку вакансій Data Science в Україні, 2020 рік. URL: https://apps.ucu.edu.ua/articles-and-research/data-science-job-market-2020/ (дата звернення: 03.05.2023).
16. CMS and Why Should you Care? ULR: https://blog.hubspot.com/blog/tabid/6307/bid/7969/what-is-a-cms-and-why-should-you-care.aspx (дата звернення: 03.05.2023).
17. Ларс Хельгенсон. CRM for Dummies, 2017. С. 31-36.
18. Леонард Кауфман. Finding Groups in Data: An Introduction to Cluster Analysis, 1990. С. 122-132.
19. Кьортіс Міллер. Training Systems Using Python Statistical Modeling, 2019. С. 53-66.
20. Андреас Мюллер. Introduction to Machine Learning with Python, 2017. С. 23-22.
21. How Applying Data Science in E-Commerce Will Boost Online Sales. URL: https://medium.com/@mygreatlearning/how-applying-data-science-in-e-commerce-will-boost-online-sales-ac42239afa91 (дата звернення: 03.05.2023).
22. Ракеш Агравал. Mining Association Rules between Sets of Items in Large Databases, 1994. С. 6-8.
23. A Guide to Association Rule Mining. URL: https://towardsdatascience.com/a-guide-to-association-rule-mining-96c42968ba6 (дата звернення: 03.05.2023).
24. Пітер Д. Ф. Лукас. Principles of Expert Systems, 1991. С. 131-137.
25. Тревор Хейсті. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2018. С. 88-95.
26. Introduction to CRISP-DM. URL: https://www.ibm.com/docs/en/spss-modeler/saas?topic=dm-crisp-help-overview (дата звернення: 03.05.2023).
27. Том Мітчелл. Machine Learning, 1997. С. 20-28.
28. Педро Домінгос. The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World. 2015. С. 55-73.

29. How Data Science increased the profitability of the e-commerce industry? URL: https://www.projectpro.io/article/how-data-science-increased-the-profitability-of-the-e-commerce-industry/168 (дата звернення: 03.05.2023).
30. Data Science in E-commerce Use Cases. URL: https://softengi.com/blog/data-science-in-e-commerce-use-cases/ (дата звернення: 03.05.2023).
31. Ясер С. Абу-Мостафа, Малік Магдон-Ізмаїл, Хсуан-Тен Лін. Learning From Data, 2012. С. 14-33.
32. K-means: A Complete Introduction. URL: https://towardsdatascience.com/k-means-a-complete-introduction-1702af9cd8c#:~:text=K%2Dmeans%20is%20an%20unsupervised%20clustering%20algorithm%20designed%20to%20partition,classifies%20them%20together%20into%20clusters (дата звернення: 03.05.2023).
33. KNN Algorithm: When? Why? How? URL: https://towardsdatascience.com/knn-algorithm-what-when-why-how-41405c16c36f (дата звернення: 03.05.2023).
34. Стівен Скієна. The Data Science Design Manual, 2017. С. 51-61.
35. Apriori Algorithm for Association Rule Learning. URL: https://towardsdatascience.com/apriori-algorithm-for-association-rule-learning-how-to-find-clear-links-between-transactions-bf7ebc22cf0a (дата звернення: 03.05.2023).
36. The FP Growth algorithm. URL: https://towardsdatascience.com/the-fp-growth-algorithm-1ffa20e839b8 (дата звернення: 03.05.2023).
37. Нейт Сільвер. The Signal and the Noise: Why So Many Predictions Fail – But Some Don't, 2012. С. 13-23.
38. Джон Форман. Data Smart: Using Data Science to Transform Information into Insight, 2013. С. 21-32.
39. Кріс Албон. Machine Learning with Python Cookbook, 2018. С. 43-49.
40. Джуда Філліпс. Ecommerce Analytics, 2017. С. 13-15.

ДОДАТКИ

ДОДАТОК А

Код роботи з методом k-середніх на прикладі датасету з даними покупців

```
#Візьмемо колонки що нас цікавлять, а саме прибуток покупця та оцінку його витрат.
X = dataset.iloc[:, [3,4]].values
from sklearn.cluster import KMeans
wcss=[]

#допустимо, що максимальна кількість кластерів буде = 10
###Отримуємо максимум кластерів

for i in range(1,11):
    kmeans = KMeans(n_clusters= i, init='k-means++', random_state=0)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)

#Візуалізуємо "Метод ліктя" для того, щоб дізнатися, яка кількість кластерів буде оптимальною
plt.plot(range(1,11), wcss)
plt.title('Метод ліктя')
plt.xlabel('кількість кластерів')
plt.ylabel('сумма квадратів відстані між точками')
plt.show()

kmeansmodel = KMeans(n_clusters= 5, init='k-means++', random_state=0)
y_kmeans= kmeansmodel.fit_predict(X)
```

```
#Візуалізуємо всі кластери

plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'red', label = 'Cluster 1')

plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'blue', label = 'Cluster 2')

plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'green', label = 'Cluster 3')

plt.scatter(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], s = 100, c = 'cyan', label = 'Cluster 4')

plt.scatter(X[y_kmeans == 4, 0], X[y_kmeans == 4, 1], s = 100, c = 'magenta', label = 'Cluster 5')

plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s = 300, c = 'yellow', label = 'Centroids')

plt.title('Кластери клієнтів')

plt.xlabel('Annual Income (k$)')

plt.ylabel('Spending Score (1-100)')

plt.legend()

plt.show()


cluster_map = pd.DataFrame()

cluster_map['CustomerID'] = dataset["CustomerID"]

cluster_map['cluster'] = y_kmeans


### Пояснення моделі

#Cluster 1 (Red Color) -> високий прибуток, але мало тратять

#cluster 2 (Blue Colr) -> середній прибуток та витрати

#cluster 3 (Green Color) -> високий прибуток та високі витрати. Є цільовою аудиторією. Для таких клієнтів є доцільним зробити

#рекламну розсилку по пошті та/або іншими засобами зв'язку

#cluster 4 (cyan Color) -> мало заробляють, але багато тратять
```

```
#Cluster 5 (magenta Color) -> мало заробляють, та мало витрачають


cluster_map.head(10)
```

ДОДАТОК Б

Код роботи з методом k-найближчого сусіда

```
books_features = pd.concat([df['Ratings_Dist'].str.get_dummies(sep=","),
df['average_rating'], df['ratings_count']], axis=1)

min_max_scaler = MinMaxScaler()

books_features = min_max_scaler.fit_transform(books_features)

np.round(books_features, 2)


model = neighbors.NearestNeighbors(n_neighbors=6, algorithm='ball_tree')

model.fit(books_features)

distance, indices = model.kneighbors(books_features)


def get_index_from_name(name):
    return df[df["title"]==name].index.tolist()[0]


all_books_names = list(df.title.values)


def get_id_from_partial_name(partial):
    for name in all_books_names:
        if partial in name:
            print(name,all_books_names.index(name))


def print_similar_books(query=None,id=None):
```

```
    if id:
        for id in indices[id][1:]:
            print(df.iloc[id]["title"])
    if query:
        found_id = get_index_from_name(query)
        for id in indices[found_id][1:]:
            print(df.iloc[id]["title"])

## Виклик

print_similar_books("The Catcher in the Rye")
print_similar_books("The Hobbit or There and Back Again")
get_id_from_partial_name("Harry Potter and the ")
```

ДОДАТОК В

Код роботи методу навчання асоціативним правилам

```
from fpgrowth_py import fpgrowth
import numpy as np
import pandas as pd
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
import time

data=pd.read_csv('data.csv')
data['GroupPrice']=data['Quantity']*data['UnitPrice']
data=data.dropna()
```

```
print('Розмірність датасету: ', data.shape)
print('---------')
data.head()

liste= data['StockCode'].unique()
stock_to_del=[]
for el in liste:
    if el[0] not in ['1','2','3','4','5','6','7','8','9','10']: # продукти що відносяться до подарунків.
        stock_to_del.append(el)

data=data[data['StockCode'].map(lambda x: x not in stock_to_del)] # видалити ці подарунки

basket = data.groupby(['InvoiceNo','CustomerID']).agg({'StockCode': lambda s: list(set(s))}) # згрупувати по чеку

print('Розмірність нового датасету: ', basket.shape)
print('----------')
basket.head()

a=time.time()
freqItemSet, rules = fpgrowth(basket['StockCode'].values, minSupRatio=0.005, minConf=0.3)
b=time.time()
print('час на виконання: ',b-a, ' s.')
print('Кількість згенерованих правил:', len(rules))

association=pd.DataFrame(rules,columns =['basket','next_product','proba'])
```

```
association=association.sort_values(by='proba',ascending=False)

print('Розмірність таблиці асоціацій: ', association.shape)

association.head(10)

def compute_next_best_product(basket_el):

    for k in basket_el: # for each element in the consumer basket

        k={k}

        if len(association[association['basket']==k].values) !=0: # if we find a corresponding association in the fp growth table

    next_pdt=list(association[association['basket']==k]['next_product'].values[0])[0] # we take the consequent product

            if next_pdt not in basket_el : # Запевняємося, що покупець раніше не купував цей товар

                proba=association[association['basket']==k]['proba'].values[0] # Знайти ймовірність асоціації

                return(next_pdt,proba)

    return(0,0) # return (0,0) якщо нічого не знайдено
```

… … … … … … …