**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

Харківський національний університет імені В. Н. Каразіна

Факультет математики і інформатики

Кафедра теоретичної та прикладної інформатики

# Кваліфікаційна робота

## бакалавр

на тему:

“Використання методів машинного навчання у проектах Business Intelligence”

Виконали: студентки 4 курсу, групи
МФ-41 спеціальність 122
«Комп’ютерні науки»
освітньо-професійна програма
«Інформатика»
Приходченко С.Р.
Зима М.А.

_____

Керівник: викл. Панченко А. С.
Рецензент: викл. Дейнега О. А.

Харків – 2024

ЗМІСТ

# ВСТУП

У сучасному швидкоплинному бізнес-середовищі організації все частіше звертаються до передових технологій, щоб отримати конкурентну перевагу. Серед цих технологій штучний інтелект (ШІ) змінив правила гри, революціонізувавши спосіб ведення бізнесу та прийняття рішень. Однією зі сфер, де ШІ має величезний потенціал, є Business intelligence, який передбачає збір, аналіз і представлення даних для спрощення прийняття рішень. Інтеграція штучного інтелекту в BI-проекти відкриває нові можливості, автоматизує процеси та підвищує загальну ефективність прийняття рішень. У цій дипломній роботі розглядаються підходи до використання штучного інтелекту в BI-проектах, аналізуються наявні рішення, досліджується поточний стан галузі, практичне значення такої інтеграції, а також потенційні переваги.

**Актуальність цієї теми** полягає у тому, що технології штучного інтелекту нині перебувають у піковому стані розвитку, що обумовлює важливість і швидкість появи нових підходів у їх застосуванні.

Традиційні підходи до аналізу бізнесу часто не встигають за обсягом, різноманітністю та швидкістю даних у сучасному бізнес-середовищі. Штучний інтелект з його здатністю вчитися на шаблонах, виявляти аномалії та робити прогнози пропонує потужне рішення цієї проблеми.

**Практичне значення** дослідження впровадження методів штучного інтелекту у розробці BI має вагому цінність, адже може призвести до більш ефективного використання даних бізнес-системами, підвищення якості прийняття рішень та виведення конкурентоспроможність компаній на новий рівень. У ході імплементації даного дослідження на практиці можуть бути застосовані наступні переваги:

1. аналітика даних відбувається через автоматизовані процеси збору, обробки та аналізу значних обсягів даних, що суттєво впливає на швидкість виявлення тенденцій та прийняття рішень;

2. прогнозування та передбачення майбутніх подій у ринковому середовищі знаходяться на вищому етапі розвитку, завдяки використанню методів ШІ, що допомагає бізнесу швидше адаптуватися до змін;

3. персоналізація суттєво допомагає зрозуміти потреби конкретного клієнта, засновуючись на унікальних проблемах та потребах бізнесу; виявлення аномалій також може сприяти більш детальному аналізу даних та виявленню проблемних зон та точок росту для бізнесу;

4. оптимізація операцій є суттєвою перевагою використання методів ШІ, так як допомагає у доцільному спрямуванні вже існуючих ресурсів, задля нарощування темпу розвитку та фіксуванні позитивного патерну бізнесу;

5. отримання досвіду, який може конвертуватися у час розробки подібних рішень, що підвищить конкурентоспроможність на ринку праці.

**Основна мета** дослідити практичні методи впровадження штучного інтелекту в проектах BI.

**Об`єктом** даного дослідження є фреймворк Plotly Dash, та бібліотеки Streamlit, Numpy і Sklearn.

**Предметом** цього дослідження є підходи до використання моделей штучного інтелекту в проектах BI.

# РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ

## 1.1 Поточна ситуація та постановка проблеми

Штучний інтелект (ШІ) та бізнес-інтелект (БІ) стали двома вагомими аспектами розвитку технологій останніх років, які змінили спосіб роботи організацій та прийняття рішень. Історія штучного інтелекту бере свій початок у 1950-х роках, коли цей термін вперше вжив Джон Маккарті на конференції в Дартмутському коледжі. З того часу штучний інтелект зазнає постійного значного розвитку - від ранніх систем, заснованих на правилах, до більш досконалих методів машинного навчання та глибокого навчання сьогодні. Аналогічно, аналітика бізнесу еволюціонувала від простих інструментів звітності та візуалізації даних до складних платформ, які використовують розширену аналітику та можливості штучного інтелекту для надання дієвих інсайтів[2].

Бізнес-орієнтований інтелект - це процеси, технології та інструменти, що використовуються для збору, інтеграції, аналізу та представлення даних для підтримки прийняття рішень в організаціях. Основна мета BI - перетворити необроблені дані на значущі ідеї, які можуть стати рушійною силою бізнес-стратегій і підвищити операційну ефективність. BI охоплює широкий спектр діяльності, включаючи зберігання даних, інтелектуальний аналіз даних, звітність, інформаційні панелі та розширену аналітику. Зі зростанням обсягів даних у цифрову епоху аналітика стає все більш важливою для організацій, щоб залишатися конкурентоспроможними та приймати обґрунтовані рішення.

Інтеграція штучного інтелекту в аналітику бізнесу відкрила для організацій нові можливості для вилучення цінності зі своїх даних. Методи ШІ, такі як машинне навчання, обробка природної мови та комп'ютерний зір, можуть автоматизувати та вдосконалити різні аспекти процесу аналітики бізнесу - від попередньої обробки даних і розробки функцій до прогнозного моделювання та

виявлення аномалій. Аналітика на основі штучного інтелекту може допомогти організаціям виявити приховані закономірності, спрогнозувати майбутні тенденції та оптимізувати бізнес-процеси таким чином, що раніше було неможливо з традиційними підходами аналізу наявних даних.

Однак впровадження штучного інтелекту в бізнес-аналітиці не позбавлене певних труднощів. Однією з основних проблем є доступність і якість даних. Алгоритми штучного інтелекту потребують великих обсягів високоякісних маркованих даних для навчання та оптимізації своїх моделей. Багато організацій стикаються з проблемами, пов'язаними з розрізненими даними, невідповідностями та прогалинами в інфраструктурі даних, які можуть перешкоджати ефективності BI-ініціатив на основі штучного інтелекту. Вирішення цих проблем вимагає значних інвестицій в управління поточною інформацією, їх інтеграцію та управління якістю.

Ще однією проблемою є брак кваліфікованих кадрів у сфері ШІ та BI. Розробка та розгортання рішень штучного інтелекту вимагає поєднання технічної експертизи в галузі науки про дані, машинного навчання та програмної інженерії, а також знань предметної області в конкретному бізнес-контексті. Організації часто стикаються з нестачею фахівців з такими навичками, що призводить до дефіциту кадрів, який сповільнює впровадження ШІ в BI-проекти. Інвестиції в програми навчання та підвищення кваліфікації, а також співпраця з навчальними закладами та дослідницькими центрами можуть допомогти подолати цей дефіцит фахівців[3].

Незважаючи на ці виклики, потенційні переваги штучного інтелекту в аналітиці бізнесу є значними. Штучний інтелект може автоматизувати рутинні завдання, звільняючи аналітиків-людей до зосередження на більш стратегічній і творчій роботі. Він має здатність виявляти закономірності та ідеї, які можуть бути пропущені аналітиками, що призводить до більш точного і своєчасного прийняття рішень. BI на основі штучного інтелекту також може забезпечити предиктивну аналітику, що дозволяє організаціям передбачати майбутні

тенденції та вживати проактивних заходів для використання можливостей та/або зменшення ризиків.

Майбутнє штучного інтелекту в аналітиці бзнесу виглядає багатообіцяючим завдяки постійному вдосконаленню технологій штучного інтелекту та зростаючій доступності даних. Однак реалізація повного потенціалу штучного інтелекту в BI вимагає комплексного підходу, що враховує технічні, організаційні та етичні проблеми. Організаціям необхідно інвестувати в надійну інфраструктуру даних, створювати міжфункціональні команди з різноманітними навичками та впроваджувати систему управління, щоб забезпечити відповідальне та ефективне використання ШІ в ініціативах бізнес-орієнтованого інтелекту.

У сучасному бізнес-середовищі розуміння поведінки, уподобань та дистрибуції клієнтів стало критично важливим аспектом прийняття стратегічних рішень та позиціонування на ринку. Здатність ефективно аналізувати та візуалізувати дані про клієнтів стала ключовим фактором диференціації для організацій, які прагнуть отримати конкурентну перевагу та надавати персоналізований досвід. Недавні дослідження підкреслили важливість аналізу сегментації та дистрибуції клієнтів для проведення цільових маркетингових кампаній, оптимізації розподілу ресурсів та підвищення загальної ефективності бізнесу. Дослідники вивчали різні методи і методології для вилучення значущої інформації з даних про клієнтів, починаючи від традиційних статистичних методів і закінчуючи передовими алгоритмами машинного навчання. Застосування методів кластеризації, таких як кластеризація за методом K-середніх, привернуло значну увагу завдяки своїй здатності ідентифікувати окремі сегменти клієнтів на основі шаблонів подібності [5]. Використовуючи ці методи, організації можуть виявити приховані закономірності, ідентифікувати високоцінні сегменти клієнтів і відповідно адаптувати свої пропозиції [6]. Крім того, інтеграція інструментів візуалізації даних та інтерактивних дашбордів революціонізувала спосіб передачі інформації про клієнтів та реагування на неї [7]. Такі платформи, як

Streamlit, стали потужними інструментами для створення інтуїтивно зрозумілих та інтерактивних додатків для візуалізації даних, що дозволяють особам, які приймають рішення, досліджувати та взаємодіяти з клієнтськими даними в режимі реального часу [8]. Поєднання аналізу та візуалізації даних дозволило організаціям приймати рішення на основі даних, оптимізувати стратегії залучення клієнтів і стимулювати зростання бізнесу [9]. Однак, незважаючи на досягнення в аналізі клієнтських даних, деякі проблеми залишаються. Зростання обсягу, різноманітності та швидкості клієнтських даних створює значні перешкоди з точки зору якості даних, інтеграції та конфіденційності. Потреба в надійних системах управління даними та етичних міркуваннях стала першочерговою для забезпечення відповідального використання клієнтських даних. Крім того, динамічна природа уподобань та поведінки споживачів вимагає постійного моніторингу та адаптації методів аналізу, щоб залишатися актуальними та ефективними.

Дане дослідження має на меті вирішити ці проблеми, пропонуючи комплексну систему для аналізу та візуалізації розподілу клієнтів. Використовуючи найсучасніші алгоритми кластеризації та інтерактивні методи візуалізації, представлена робота має на меті надати компаніям практичні поради щодо оптимізації їхніх стратегій залучення клієнтів. Запропонована методологія включає методи попередньої обробки даних для обробки пропущених значень, викидів і невідповідностей даних, забезпечуючи цілісність і надійність аналізу. Крім того, дослідження вивчає потенціал інтеграції додаткових джерел даних, таких як дані соціальних мереж та відгуки клієнтів, для збагачення аналізу та надання цілісного уявлення про поведінку клієнтів. Значення цього дослідження полягає в тому, що воно здатне подолати розрив між теоретичними концепціями та практичним застосуванням, надаючи організаціям можливість використовувати дані клієнтів для прийняття стратегічних рішень та отримання конкурентних переваг. Надаючи надійну і масштабовану основу для аналізу розподілу клієнтів, це дослідження має на

меті зробити внесок у зростаючий обсяг знань в області аналізу користувачів бізнесів і прийняття рішень на основі даних.

Як наслідок, останніми роками спостерігається активний розвиток нових підходів до кластеризації, зокрема, заснованих на нечітких множинах, нейронних мережах та еволюційних алгоритмах. Ці методи демонструють покращену стійкість до шуму та здатність обробляти нелінійні та гетерогенні дані. Тим не менш, вони також мають свої недоліки, включаючи високу обчислювальну складність і складність у налаштуванні параметрів. Окрім самих алгоритмів кластеризації, важливу роль відіграє попередня обробка даних, яка охоплює очищення, нормалізацію та кодування категоріальних змінних. Неправильна підготовка даних може призвести до спотворення результатів і зниження якості кластеризації. Іншим актуальним напрямом досліджень є візуалізація результатів кластеризації, що сприяє кращій інтерпретації отриманих груп та виявленню прихованих закономірностей. Сучасні бібліотеки візуалізації даних, такі як Plotly, Bokeh та D3.js, пропонують багатий набір інструментів для створення інтерактивних та інформативних графіків. Нарешті, важливо зазначити, що ефективність методів кластеризації сильно залежить від конкретної предметної області та характеристик аналізованих даних. Тому дуже важливо ретельно оцінювати застосовність різних підходів і проводити всебічну перевірку результатів з використанням відповідних метрик якості.

Методи сегментації та кластеризації клієнтів широко вивчені в літературі, при цьому для вирішення проблем, пов'язаних з цим завданням, пропонуються різні підходи та методології. Алгоритм k-середніх, запроваджений у 1967 році, залишається одним з найпоширеніших методів кластеризації завдяки своїй простоті та ефективності. Однак він має ряд обмежень, серед яких чутливість до викидів, необхідність попереднього визначення кількості кластерів та складність роботи з неглобулярними формами кластерів. Щоб подолати ці недоліки, дослідники вивчали альтернативні алгоритми кластеризації, такі як ієрархічна кластеризація, просторова кластеризація на основі щільності

(DBSCAN) [5] та кластеризація на основі моделей . Ці методи пропонують підвищену надійність і гнучкість, але часто ціною підвищення обчислювальної складності та необхідності налаштування параметрів.

Огляд літератури з аналізу та візуалізації дистрибуції клієнтів розкриває багате полотно досліджень, що охоплюють різні сфери, включаючи маркетинг, науку про дані та інформаційні системи. В останніх дослідженнях постійно підкреслюється зростаюча важливість розуміння поведінки клієнтів та моделей дистрибуції . Дослідники вивчали теоретичні основи та практичне застосування методів сегментації клієнтів, підкреслюючи їх значення для розробки цільових маркетингових стратегій та оптимізації розподілу ресурсів. Поява великих даних і передової аналітики ще більше розвинула сферу аналізу групування клієнтів, дозволивши організаціям використовувати величезні обсяги даних про користувачів бізнесу для прийняття стратегічних рішень. Численні дослідження вивчали ефективність різних алгоритмів кластеризації, таких як K-середні, ієрархічна кластеризація та кластеризація на основі щільності, у визначенні окремих сегментів клієнтів на основі демографічних, поведінкових та трансакційних атрибутів .

Ці алгоритми були застосовані до різноманітних наборів даних, починаючи від транзакцій електронної комерції і закінчуючи взаємодією в соціальних мережах, демонструючи їхню універсальність і надійність. Більше того, інтеграція методів машинного навчання, таких як нейронні мережі та машини опорних векторів, ще більше підвищила точність і масштабованість моделей сегментації клієнтів. У літературі також підкреслюється критична роль попередньої обробки даних та інженерії ознак у забезпеченні якості та релевантності даних про клієнтів для аналізу. Дослідники запропонували різні методи для обробки пропущених значень, викидів і неузгодженостей даних, а також методи відбору ознак і зменшення розмірності. Підкреслено важливість нормалізації та стандартизації даних для забезпечення порівнянності та інтерпретованості атрибутів клієнтів за різними шкалами та одиницями виміру. На додаток до технічних аспектів аналізу розподілу клієнтів, в літературі також

досліджуються етичні питання та питання конфіденційності, пов'язані з використанням даних про клієнтів [10]. Підкреслюється необхідність прозорих практик збору даних, інформованої згоди та безпечного зберігання даних для захисту приватності клієнтів та збереження довіри. Дослідники також обговорювали потенційні упередження та обмеження моделей сегментації клієнтів, наголошуючи на важливості регулярної перевірки та оновлення цих моделей для відображення змін у поведінці клієнтів та динаміці ринку. Візуалізація інформації про розподіл клієнтів стала важливою сферою досліджень, причому дослідження зосереджуються на розробці та оцінці інтерактивних інформаційних панелей і методів візуалізації даних. У літературі підкреслюється ефективність візуальної аналітики для полегшення дослідження даних, розпізнавання шаблонів і прийняття рішень. Дослідники запропонували різні фреймворки візуалізації та принципи дизайну для покращення зручності використання та інтерпретації візуалізацій розподілу клієнтів, що дає змогу бізнес-користувачам отримувати дієві висновки. Огляд літератури показує, що дослідження в галузі аналізу та візуалізації розподілу клієнтів динамічно розвиваються. Синтез теоретичних концепцій, практичних застосувань та етичних міркувань забезпечує міцну основу для подальших досліджень та інновацій у цій галузі. Використовуючи ідеї та найкращі практики з літератури, організації можуть використовувати силу даних про клієнтів для прийняття стратегічних рішень, оптимізації взаємодії з клієнтами та отримання конкурентних переваг на ринку.

В останні роки значна увага приділяється застосуванню методів машинного навчання та глибокого навчання для сегментації клієнтів. Підходи, засновані на картах, що самоорганізуються (SOM), нечіткій кластеризації k-середніх та нейронних мережах, продемонстрували багатообіцяючі результати у виявленні складних патернів та обробці даних високої розмірності. Однак інтерпретованість цих моделей залишається проблемою, і необхідні подальші дослідження для забезпечення їх практичної застосовності. Іншим важливим аспектом сегментації клієнтів є вибір релевантних ознак та етапів попередньої

обробки даних. Такі методи, як аналіз головних компонент (PCA), алгоритми відбору ознак та методи інтерполяції даних  були використані для підвищення якості та ефективності моделей кластеризації. Крім того, було досліджено інтеграцію знань про предметну область та експертних висновків для покращення інтерпретації та практичного застосування результатів сегментації. Незважаючи на значні дослідницькі зусилля, сегментація клієнтів залишається складним завданням, оскільки вона вимагає балансування між кількома цілями, такими як якість кластера, інтерпретованість та обчислювальна ефективність. Крім того, природа даних про клієнтів, що швидко змінюється, і поява нових джерел даних (наприклад, соціальні мережі, Інтернет речей) вимагають постійного розвитку та адаптації методів сегментації для вирішення цих динамічних завдань.

### 1.2. Аналіз існуючих програм

Для того, щоб отримати комплексне уявлення про існуючі рішення для сегментації та кластеризації клієнтів, було проведено аналіз різних програм та програмних інструментів. Однією з найбільш широко використовуваних бібліотек з відкритим вихідним кодом для аналізу даних є scikit-learn [1], яка надає ряд алгоритмів кластеризації, включаючи k-середні, ієрархічну кластеризацію, DBSCAN та інші. Ці алгоритми пропонують гнучкість і масштабованість, але вимагають значного досвіду кодування та кроків попередньої обробки даних. Іншою популярною бібліотекою є Keras, яка дозволяє реалізовувати моделі кластеризації на основі нейронних мереж. Аналіз існуючих програм у сфері аналізу та візуалізації розподілу клієнтів виявляє різноманітний ландшафт програмних рішень та фреймворків. Ці програми охоплюють широкий спектр функціональних можливостей, від попередньої обробки та інтеграції даних до розширеної аналітики та інтерактивної візуалізації. Критична оцінка цих існуючих програм висвітлює їхні сильні сторони, обмеження та потенціал для вдосконалення у вирішенні проблем

аналізу розподілу клієнтів . Окрема категорія програм зосереджена на інтеграції та попередній обробці даних, що є важливими кроками у забезпеченні якості та узгодженості даних про клієнтів. Такі інструменти, як Talend, Alteryx та Informatica, надають надійні можливості інтеграції даних, дозволяючи організаціям витягувати, трансформувати та завантажувати дані про клієнтів з різних джерел.

Ці програми пропонують такі функції, як профілювання даних, очищення та збагачення даних, які допомагають виявляти та вирішувати проблеми з якістю даних. Однак складність і тривалість навчання, пов'язані з цими інструментами, часто створюють проблеми для бізнес-користувачів, які не мають великого технічного досвіду . Інша група програм зосереджена на розширеній аналітиці та методах машинного навчання для сегментації клієнтів та аналізу дистрибуції. Такі платформи, як RapidMiner, KNIME та SAS Enterprise Miner, надають комплексний набір інструментів для дослідження даних, інженерії функцій та розробки моделей. Ці програми пропонують широкий спектр алгоритмів кластеризації, таких як K-середні, ієрархічна кластеризація та карти, що самоорганізуються, які зазвичай використовуються для сегментації клієнтів . Хоча ці програми надають потужні аналітичні можливості, вони часто вимагають глибокого розуміння концепцій машинного навчання та навичок програмування, що може обмежувати їхню доступність для нетехнічних користувачів . У сфері візуалізації даних та інтерактивних дашбордів значної популярності набули такі програми, як Tableau, Power BI та QlikView . Ці інструменти дозволяють користувачам створювати візуально привабливі та інтерактивні дашборди, які полегшують дослідження даних та виявлення інсайтів. Вони пропонують широкий спектр типів діаграм, карт та інших компонентів візуалізації, які можна налаштувати для ефективного представлення моделей розподілу клієнтів . Однак ефективність цих візуалізацій значною мірою залежить від здатності користувача вибрати відповідні візуальні кодування та принципи дизайну, що може вимагати спеціальної підготовки та досвіду. Незважаючи на прогрес в існуючих

програмах, у сфері аналізу та візуалізації розподілу клієнтів зберігається низка проблем та обмежень. Інтеграція розрізнених джерел даних, обробка неструктурованих і напівструктурованих даних, а також потреба в аналітиці в режимі реального часу є одними з ключових областей, які потребують подальших досліджень і розробок. Крім того, інтерпретація та передача інформації, отриманої в результаті аналізу розподілу клієнтів, залишаються значними проблемами, особливо для нетехнічних зацікавлених сторін. Аналіз існуючих програм підкреслює потребу в більш зручних і доступних рішеннях, які подолають розрив між розширеною аналітикою та інтуїтивно зрозумілою візуалізацією.

Майбутні дослідження повинні бути зосереджені на розробці фреймворків і методологій, які дозволяють безперешкодно інтегрувати попередню обробку даних, аналітику та візуалізацію, забезпечуючи при цьому керовані робочі процеси та інтерактивні інтерфейси для бізнес-користувачів . Крім того, включення зрозумілих методів штучного інтелекту та принципів дизайну, орієнтованих на користувача, може підвищити інтерпретованість і достовірність результатів аналізу дистрибуції клієнтів. На закінчення, аналіз існуючих програм для аналізу та візуалізації розподілу клієнтів виявляє багату екосистему інструментів і фреймворків, кожен з яких має свої сильні та слабкі сторони. Хоча ці програми досягли значних успіхів у наданні організаціям можливості отримувати інформацію з даних про клієнтів, залишається багато можливостей для вдосконалення з точки зору зручності використання, доступності та інтерпретації. Вирішуючи ці проблеми та використовуючи уроки, отримані з існуючих програм, дослідники та практики можуть розробити більш ефективні та орієнтовані на користувача рішення для аналізу та візуалізації дистрибуції клієнтів.

Однак ці моделі можуть бути складними для інтерпретації і потребують ретельного налаштування для досягнення оптимальної продуктивності. У сфері комерційного програмного забезпечення IBM SPSS Modeler та SAS Enterprise Miner пропонують комплексні набори інструментів для інтелектуального

аналізу даних та кластеризації. Хоча ці платформи надають зручні інтерфейси та розширені можливості візуалізації, вони часто мають значну вартість і їм може бракувати гнучкості рішень з відкритим вихідним кодом. Tableau та Power BI широко використовуються для візуалізації даних та інформаційних панелей, пропонуючи інтерактивні візуалізації та інтеграцію з різними джерелами даних. Однак їхні можливості кластеризації обмежені, і вони зазвичай покладаються на сторонні інструменти або користувацькі скрипти для більш просунутого аналізу. Кілька хмарних платформ, таких як Google Cloud AI Platform, Amazon SageMaker та Microsoft Azure Machine Learning, стали потужними рішеннями для сегментації та кластеризації клієнтів. Ці платформи пропонують масштабовані обчислювальні ресурси, автоматизоване розгортання моделей та інтеграцію з широким спектром джерел даних. Тим не менш, вони можуть мати крутіші криві навчання і можуть бути дорогими для невеликих організацій. Крім того, для сегментації клієнтів було розроблено кілька спеціалізованих інструментів, таких як Kissmetrics [10], RJMetrics та Woopra. Ці інструменти часто інтегруються з існуючими системами управління взаємовідносинами з клієнтами (CRM) і надають індивідуальні візуалізації та рекомендації для сегментації клієнтів. Однак їхня функціональність може бути обмежена конкретними випадками використання, і їм може бракувати гнучкості та можливостей кастомізації, які притаманні більш універсальним інструментам. Важливо зазначити, що вибір найбільш підходящої програми або інструменту залежить від різних факторів, включаючи розмір і складність набору даних, конкретні вимоги та цілі аналізу, доступні обчислювальні ресурси та досвід користувачів [5]. Ретельна оцінка цих факторів необхідна для забезпечення ефективного впровадження та використання методів сегментації та кластеризації споживачів.

### 1.3. Опис історії розвитку штучного інтелекту та BI

Розвиток штучного інтелекту (ШІ) та бізнес-інтелекту (БІ) формувався десятиліттями досліджень, технологічного прогресу та мінливих потреб бізнесу. Коріння штучного інтелекту можна простежити в 1940-х і 1950-х роках, коли першопрохідці в галузі комп'ютерних наук і математики заклали основи інтелектуальних машин. У 1950 році Алан Тюрінг запропонував тест Тюрінга - метод оцінки здатності машини демонструвати розумну поведінку, яку неможливо відрізнити від людської. Ця фундаментальна робота викликала інтерес до можливості створення машин, які могли б думати і міркувати, як люди[11].

У наступні десятиліття дослідження ШІ були зосереджені на розробці систем, заснованих на правилах, експертних систем і символічних міркувань. Дослідники вивчали різні підходи, такі як дерева рішень, логічне програмування та представлення знань, щоб імітувати процеси прийняття рішень, подібні до людських. Однак ці ранні системи штучного інтелекту були обмежені їхньою залежністю від явних правил та інженерії знань, що робило їх крихкими і складними для масштабування[12].

У 1980-х і 1990-х роках з'явилося машинне навчання як домінуюча парадигма в ШІ. Алгоритми машинного навчання, такі як дерева рішень, нейронні мережі та машини опорних векторів, дозволили системам навчатися на основі даних і з часом покращувати свою продуктивність. Цей зсув у бік підходів, заснованих на даних, революціонізував ШІ, зробивши його більш адаптивним і застосовним до ширшого кола проблем.

Паралельно з цим розвивалася сфера бізнес-аналітики, щоб задовольнити зростаючу потребу в прийнятті рішень на основі даних в організаціях. BI охоплювала набір процесів, технологій та інструментів, які перетворювали необроблені дані на змістовні та дієві інсайти. Перші системи BI були зосереджені на зберіганні даних, звітності та аналітичній обробці в режимі онлайн (OLAP). Ці системи дозволяли компаніям консолідувати та аналізувати величезні обсяги структурованих даних, допомагаючи їм виявляти тенденції, закономірності та можливості для вдосконалення.

Наприкінці 1990-х - на початку 2000-х років, коли обсяги та різноманітність даних різко зросли, стала очевидною потреба в більш досконалій аналітиці та інсайтах у режимі реального часу. Поява технологій великих даних, таких як бази даних Hadoop і NoSQL, дозволила організаціям зберігати, обробляти та аналізувати величезні обсяги структурованих і неструктурованих даних. Це призвело до розвитку нових аналітичних методів, таких як інтелектуальний аналіз даних, прогнозне моделювання та текстова аналітика, що дозволило компаніям виявляти приховані закономірності та інсайти у своїх даних.

Конвергенція AI та BI в останні роки змінила ландшафт прийняття рішень на основі даних. Алгоритми машинного навчання, мережі глибокого навчання та методи обробки природної мови були інтегровані в BI-платформи, що уможливило більш складний та автоматизований аналіз даних. Ці BI-системи на основі штучного інтелекту можуть обробляти величезні обсяги даних, виявляти аномалії, прогнозувати майбутні тенденції та надавати дієві рекомендації особам, які приймають рішення.

Поширення хмарних обчислень і моделей "програмне забезпечення як послуга" (SaaS) ще більше демократизувало доступ до можливостей штучного інтелекту і бізнес-аналітики. Хмарні BI-платформи пропонують масштабовані, гнучкі та економічно ефективні рішення для організацій будь-якого розміру, усуваючи необхідність значних початкових інвестицій в апаратну та програмну інфраструктуру. Це дозволило навіть малим і середнім підприємствам використовувати передову аналітику та ідеї на основі штучного інтелекту для розвитку свого бізнесу.

Більше того, дедалі ширше впровадження інструментів самообслуговування BI дало змогу бізнес-користувачам досліджувати та аналізувати дані самостійно, не покладаючись на IT-команди чи аналітиків даних. Ці зручні інтерфейси в поєднанні з можливостями пошуку даних за допомогою штучного інтелекту та запитів природною мовою полегшили

нетехнічним користувачам задавати питання, візуалізувати дані та отримувати значущі інсайти.

Сьогодні штучний інтелект і бізнес-аналітика продовжують розвиватися і формувати майбутнє прийняття рішень на основі даних. Інтеграція технологій штучного інтелекту, таких як машинне навчання, глибинне навчання та обробка природної мови, в BI-платформи відкрила нові можливості для автоматизації складного аналізу, виявлення прихованих закономірностей та надання прогнозних і рекомендаційних інсайтів. Оскільки організації генерують і збирають дедалі більші обсяги даних, роль штучного інтелекту та бізнес-аналітики у вилученні цінності та досягненні успіху в бізнесі стала як ніколи важливою.

Забігаючи наперед, можна сказати, що майбутнє штучного інтелекту та аналітики обіцяє ще більш трансформаційні зміни. Очікується, що досягнення в таких галузях, як пояснювальний ШІ, навчання з підкріпленням і навчання з перенесенням, підвищать інтерпретованість, адаптивність і узагальнюючі можливості систем ШІ. Конвергенція ШІ з іншими новими технологіями, такими як Інтернет речей (IoT), блокчейн і периферійні обчислення, ймовірно, створить нові можливості для аналізу даних у режимі реального часу, децентралізованого прийняття рішень і інтелектуальної автоматизації.

У міру того, як штучний інтелект і бізнес-аналітика продовжують розвиватися, організації повинні прийняти культуру, засновану на даних, інвестувати в правильні технології і таланти, а також створити надійну систему управління, щоб використовувати весь потенціал цих трансформаційних технологій. Ефективно використовуючи штучний інтелект і бізнес-аналітику, компанії можуть отримати конкурентну перевагу, стимулювати інновації та приймати більш обґрунтовані і своєчасні рішення у все більш складному і динамічному бізнес-середовищі.

Історія розвитку штучного інтелекту та бізнес-аналітики була позначена значними віхами, технологічними проривами та безперервною еволюцією, спрямованою на задоволення мінливих потреб організацій. Від перших днів

систем, заснованих на правилах, до нинішньої ери BI-платформ на основі штучного інтелекту, ця галузь стала свідком значного прогресу в забезпеченні прийняття рішень на основі даних. ШІ та бізнес-аналітика продовжують розвиватися і зближуватися, вони мають величезний потенціал для трансформації способів ведення бізнесу, впровадження інновацій та створення цінності в цифрову епоху.

### 1.4. Постановка завдання

1. Визначити предметну область для реалізації BI-проекта.
2. Дослідити предметну область, визначити основні проблеми, які можна розв’язати за допомогою використання інтерактивного відображення агрегації даних
3. Провести порівняльний аналіз BI-інструментів з точки зору можливості та легкості інтеграції з моделями машинного навчання
4. Безпосередня реалізація BI-проекту, який буде поєднувати у собі інтерактивні відображення даних (дашборди) та прогнози, побудовані з використанням моделей машинного навчання
5. Аналіз отриманих результатів

Очікуваним результатом є створення BI-проекту з інтегрованими методами машинного навчання, який прозоро дає рекомендації в цій предметній галузі, а також дозволить підвищити точність та оперативність прийняття рішень, поліпшити ключові показники ефективності бізнесу.

## РОЗДІЛ 2. ПЕРЕЛІК ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

### 2.1. Dash: Інструментарій для веб-візуалізації та аналізу даних

У сфері візуалізації та аналітики даних вибір правильного інструментарію має вирішальне значення для розробки потужних та інтерактивних додатків. Серед різних доступних варіантів Dash став переконливим вибором для науковців, аналітиків і розробників, які прагнуть легко і гнучко створювати веб-додатки, керовані даними. У цьому розділі розглядаються причини, чому Dash є найкращим інструментарієм, висвітлюються його сильні сторони, розглядаються потенційні недоліки та демонструється його придатність для широкого спектру проектів з візуалізації та аналізу даних.

| Особливості | PowerBI | Tableau | Plotly Dash |
|---|---|---|---|
| Кастомізація | Можливості кастомізації менш розвинуті | Добре кастомізовані інформаційні панелі | Висока кастомізація та контроль над додатками |
| Можливість впровадження методів штучного інтелекту | Є інструментом з вбудованими ШІ можливостями | Пропонує можливості для ШІ з інтеграцією R, Python та Einstein Discovery | Є найбільш гнучким інструментом для ШІ завдяки повній інтеграції з Python |
| Інтеграція систем управління базами даних | Широка підтримка | Наявна | Забезпечує широку підтримку інтеграції з різними СУБД |
| UI/UX | Інтуїтивний | Високоінтуїтивний, але вимагає навчання для освоєння розширених функцій | Вимагає знання програмування |
| Інтеграція з іншими продуктами | Відмінна інтеграція з екосистемою Microsoft. | Підтримка різних джерел даних | Інтеграція з Python бібліотеками |

| | | | |
|---|---|---|---|
| Вартість | Наявна безкоштовна версія з не повним функціоналом | 14-денний безкоштовний термін повного обсягу функціоналу | Повністю безкоштовна та відкрита бібліотека |
| Гнучкість | Низька | Середня | Дуже висока |
| Обсяги даних | Ліпше для маленьких наборів даних | виграє у обробці величезних наборів даних | різноманітні обсяги наборів даних |
| Оновлення даних (репортинг) | Автоматичне | Не автоматичне | Автоматичне |
| Підтримка користувача | Є | Є, велике комьюніті, більш тверда підтримка | Оскільки Dash є проектом з відкритим кодом, спільнота активно працює над виявленням та усуненням вразливостей |
| Оперативні системи | Десктопна версія підтримується тільки Windows | Windows, macOS, Linux | Працює на будь-якій операційній системі, яка підтримує Python |
| Використання бізнесами | Від маленьких до великих | більше для великих | від маленьких до великих |
| Поріг входу | Середній | Середній | Вимагає знань програмування |

Провівши деталізований аналіз Power BI, Tableau та Plotly Dash, ми дійшли висновку, що саме останній ідеально підходить для висвітлення задач даного дослідження. У контексті веб-візуалізації та аналітики даних Dash став переконливим вибором завдяки своїм численним сильним сторонам і перевагам над альтернативними фреймворками. Щоб продемонструвати придатність та

ефективність Dash, важливо висвітлити його ключові особливості, переваги та те, як він відповідає конкретним вимогам поточного проекту.

Однією з основних причин вибору Dash є його безшовна інтеграція з екосистемою Python. Python став мовою де-факто для науки про дані, машинного навчання та аналітики завдяки своїм широким бібліотекам, фреймворкам та інструментам. Dash використовує можливості Python, надаючи декларативний та реактивний фреймворк для створення інтерактивних веб-додатків. Ця тісна інтеграція дозволяє розробникам безперешкодно використовувати наявні навички та бібліотеки Python, такі як Pandas, NumPy та Scikit-learn, у додатках Dash. Використовуючи знайомість та надійність Python, Dash забезпечує швидку розробку, повторне використання коду та доступ для широкого спектру можливостей маніпулювання та аналізу даних.

З іншого боку Tableau є високо інтуїтивним інструментом для візуалізації даних з інтерфейсом "drag-and-drop", що робить його зручним для користувачів. Інтерфейс користувача добре продуманий і розроблений для легкого дослідження даних. Це робить Tableau підходящим як для початківців, так і для досвідчених користувачів.

Power BI, подібно до Tableau, має інтуїтивний інтерфейс "drag-and-drop" і добре інтегрується з іншими сервісами Microsoft. Це забезпечує користувачам знайомий і зручний досвід, особливо для тих, хто вже використовує продукти Microsoft Office.

Хоча Dash пропонує численні переваги, важливо враховувати потенційні недоліки та обмеження. Одним з них є крива навчання, пов'язана з Dash, особливо для тих, хто не знайомий з Python або концепціями веб-розробки. Розробники повинні мати ґрунтовне розуміння Python і комфортно працювати з веб-технологіями, такими як HTML, CSS та JavaScript. Однак, велика документація, навчальні посібники та підтримка спільноти, доступні для Dash, допомагають пом'якшити цю навчання і дозволяють розробникам швидко увійти в курс справи.

Незважаючи на ці міркування, переваги Dash значно переважають його обмеження. Його безшовна інтеграція з екосистемою Python, безкоштовне використання, прекрасні можливості обробки великої кількості даних, роблять його чудовим вибором для створення BI-проекту.

## 2.2. Системні вимоги

Для забезпечення ефективного та результативного впровадження запропонованого рішення для сегментації клієнтів, необхідно ретельно розглянути декілька системних вимог. По-перше, система повинна бути спроектована для роботи на надійній апаратній інфраструктурі, здатній справлятися з обчислювально інтенсивними завданнями, такими як великомасштабна обробка даних і просунуті алгоритми машинного навчання[10]. Це може передбачати використання ресурсів високопродуктивних обчислень (HPC), включаючи багатоядерні процесори, графічні процесори (GPU) та розподілені обчислювальні кластери, для прискорення обробки даних та навчання моделей. Крім того, система повинна бути оптимізована для ефективного управління пам'яттю та використання сховища, оскільки набори даних клієнтів можуть бути надзвичайно великими і вимагати значного обсягу пам'яті. Архітектура програмного забезпечення повинна бути модульною та розширюваною, що дозволяє легко інтегрувати нові алгоритми, джерела даних та компоненти візуалізації в міру розвитку системи та появи нових вимог.

Системні вимоги до платформи аналізу та візуалізації дистрибуції клієнтів мають вирішальне значення для визначення технічних специфікацій, критеріїв продуктивності та інфраструктури, необхідних для підтримки бажаних функціональних можливостей та користувацького досвіду [1]. Ці вимоги ґрунтуються на поточному стані досліджень у цій галузі, найкращих галузевих практиках та конкретних потребах організації. Вимоги до апаратного забезпечення системи повинні бути ретельно продумані, щоб забезпечити оптимальну продуктивність і масштабованість. Серверна інфраструктура

повинна мати достатню обчислювальну потужність, пам'ять і ємність для зберігання даних, щоб обробляти великі обсяги даних клієнтів і одночасні запити користувачів. Використання фреймворків розподілених обчислень, таких як Apache Hadoop або Apache Spark, дозволяє ефективно обробляти та аналізувати великі масиви даних. Система також повинна бути спроектована з можливістю горизонтального масштабування, що дозволяє додавати нові серверні вузли, щоб впоратися зі зростаючим навантаженням на дані та користувачів. Іншим важливим аспектом системних вимог є мережева інфраструктура. Система повинна мати надійне та високошвидкісне мережеве підключення, щоб забезпечити безперебійну передачу даних та їх обробку в режимі реального часу. Використання мереж доставки контенту (CDN) можна розглянути для підвищення продуктивності та доступності компонентів візуалізації, особливо для географічно розподілених користувачів. Крім того, система повинна реалізовувати надійні заходи безпеки, такі як брандмауери, системи виявлення вторгнень та віртуальні приватні мережі (VPN), щоб захистити мережу від несанкціонованого доступу та кіберзагроз. Вимоги до програмного забезпечення системи охоплюють операційну систему, систему управління базами даних, мови програмування та бібліотеки.

При виборі операційної системи слід враховувати такі фактори, як сумісність, безпека та продуктивність, серед популярних варіантів - Linux, Windows Server та macOS. Система управління базами даних повинна бути здатна обробляти структуровані та неструктуровані дані, підтримувати великі обсяги транзакцій, а також забезпечувати ефективні можливості запитів та індексування. Реляційні бази даних, такі як PostgreSQL і MySQL, а також NoSQL бази даних, такі як MongoDB і Cassandra, можуть бути розглянуті на основі конкретних вимог системи. Мови програмування та бібліотеки, що використовуються для розробки системи, слід обирати на основі їхньої придатності для аналізу даних, машинного навчання та візуалізації [15]. Python став популярним вибором завдяки своїй розгалуженій екосистемі бібліотек, таких як NumPy, Pandas та Scikit-learn, які надають потужні інструменти для

маніпулювання даними, статистичного аналізу та машинного навчання [16]. Бібліотеки JavaScript, такі як D3.js та Chart.js, широко використовуються для створення інтерактивних та візуально привабливих візуалізацій даних у веб-додатках. Система також повинна враховувати використання фреймворків обробки великих даних, таких як Apache Hadoop та Apache Spark, для обробки та аналізу великих обсягів даних. Вимоги до інтеграції системи повинні бути ретельно оцінені, щоб забезпечити безперебійну сумісність з існуючими системами та джерелами даних. Система повинна надавати API та конектори для інтеграції з системами управління взаємовідносинами з клієнтами (CRM), платформами автоматизації маркетингу та іншими відповідними корпоративними додатками. Використання стандартних форматів обміну даними, таких як JSON і XML, може полегшити інтеграцію даних і забезпечити сумісність з широким спектром систем.

Система також повинна підтримувати інтеграцію потоків даних у реальному часі, таких як стрічки соціальних мереж і дані з датчиків, щоб забезпечити актуальний аналіз і візуалізацію моделей розподілу клієнтів [22]. Продуктивність і масштабованість є критично важливими системними вимогами для забезпечення оперативності та ефективності платформи [23]. Система повинна бути спроектована для роботи з високим рівнем паралелізму, з можливістю обробки декількох запитів користувачів одночасно без шкоди для продуктивності. Використання механізмів кешування, таких як Redis або Memcached, може значно покращити час відгуку даних, до яких часто звертаються, та зменшити навантаження на внутрішні системи. Для рівномірного розподілу робочого навантаження між декількома вузлами сервера слід використовувати методи балансування навантаження, що забезпечують оптимальне використання ресурсів і запобігають утворенню вузьких місць [26]. Система також повинна бути спроектована таким чином, щоб впоратися зі зростанням обсягу даних і пристосуватися до збільшення кількості користувачів з плином часу. Важливими аспектами системних вимог є зручність використання та досвід користувачів . Інтерфейс користувача повинен бути

інтуїтивно зрозумілим, чуйним і візуально привабливим, забезпечуючи безперебійну роботу для користувачів з різним рівнем технічної підготовки. Система повинна пропонувати можливості кастомізації, дозволяючи користувачам пристосовувати візуалізації та інформаційні панелі до своїх конкретних потреб та вподобань. Використання принципів адаптивного дизайну може забезпечити доступність і зручність використання системи на різних пристроях і з різними розмірами екранів. Отже, системні вимоги до платформи аналізу та візуалізації дистрибуції клієнтів охоплюють широкий спектр технічних специфікацій, критеріїв ефективності та міркувань щодо інфраструктури .

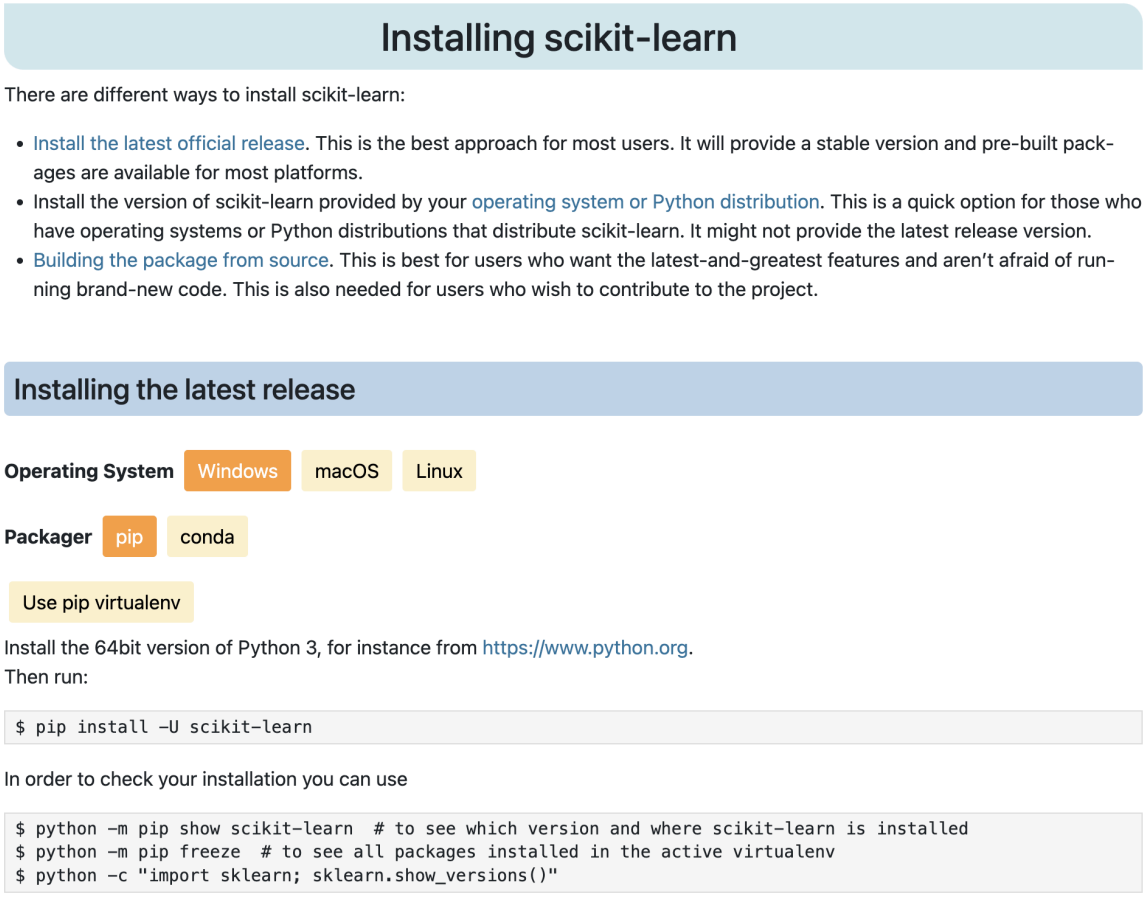
Ці вимоги ґрунтуються на поточному стані досліджень, найкращих галузевих практиках та конкретних потребах організації. Ретельно визначивши та виконавши ці системні вимоги, платформа може забезпечити надійне, масштабоване та зручне рішення для аналізу та візуалізації моделей розподілу клієнтів, що дозволить приймати рішення на основі даних та сприятиме успіху бізнесу. Система також повинна бути розроблена з урахуванням можливості масштабування, що дозволить обробляти набори даних різного розміру та складності без шкоди для продуктивності або ефективності використання ресурсів. Крім того, система повинна передбачати надійні заходи безпеки для забезпечення конфіденційності та цілісності даних клієнтів, включаючи шифрування, контроль доступу та механізми аудиту. Сумісність зі стандартними галузевими форматами даних і протоколами має важливе значення для полегшення обміну даними та інтеграції з існуючими системами бізнес-аналітики та управління даними. Інтерфейс користувача має бути інтуїтивно зрозумілим і зручним, орієнтованим як на технічні, так і на нетехнічні зацікавлені сторони, а також підтримувати різні способи взаємодії, такі як інтерфейси командного рядка, веб-панелі управління та інтерактивні візуалізації. Крім того, система повинна містити велику документацію, навчальні матеріали та допоміжні ресурси для полегшення безперешкодного впровадження та ефективного використання користувачами з різним рівнем

знань. Нарешті, система повинна відповідати найкращим галузевим практикам і стандартам розробки, тестування, розгортання та обслуговування програмного забезпечення, забезпечуючи якість коду, надійність і ремонтопридатність у довгостроковій перспективі.

### 2.3 Вимоги до програмного забезпечення

Щоб полегшити розробку і розгортання запропонованого рішення для сегментації клієнтів, необхідно встановити комплексний набір вимог до програмного забезпечення. Система повинна бути побудована на надійній та масштабованій мові програмування, такій як Python або R, які пропонують широкі бібліотеки та фреймворки для маніпулювання даними, машинного навчання та візуалізації. Зокрема, інтеграція популярних бібліотек аналізу даних, таких як Pandas, NumPy та SciPy, є важливою для ефективної обробки та попередньої обробки даних. Крім того, система повинна використовувати відомі бібліотеки машинного навчання, такі як Scikit-learn, TensorFlow або PyTorch, для реалізації широкого спектру алгоритмів кластеризації та передових методів, таких як нейронні мережі та карти, що самоорганізуються. Вимоги до програмного забезпечення для платформи аналізу та візуалізації дистрибуції клієнтів мають вирішальне значення для визначення функціональних і нефункціональних аспектів системи, забезпечення її ефективності, надійності та зручності використання. Ці вимоги є похідними від ретельного аналізу бізнес-цілей, потреб користувачів та поточного стану досліджень у цій галузі. Основною вимогою до програмного забезпечення є здатність ефективно обробляти великі обсяги даних про клієнтів. Платформа повинна бути здатна отримувати дані з різних джерел, таких як бази даних, сховища даних та API, а також виконувати необхідні завдання з очищення, трансформації та інтеграції даних. Використання технологій великих даних, таких як Apache Hadoop або Apache Spark, може уможливити розподілену обробку великих масивів даних, забезпечуючи масштабованість і продуктивність. Програмне забезпечення

також має включати надійні механізми управління якістю даних для обробки пропущених значень, викидів і невідповідностей, підтримуючи цілісність і надійність аналізу.



## Installing scikit-learn

There are different ways to install scikit-learn:

- Install the latest official release. This is the best approach for most users. It will provide a stable version and pre-built packages are available for most platforms.
- Install the version of scikit-learn provided by your operating system or Python distribution. This is a quick option for those who have operating systems or Python distributions that distribute scikit-learn. It might not provide the latest release version.
- Building the package from source. This is best for users who want the latest-and-greatest features and aren't afraid of running brand-new code. This is also needed for users who wish to contribute to the project.

### Installing the latest release

**Operating System** Windows macOS Linux

**Packager** pip conda

Use pip virtualenv

Install the 64bit version of Python 3, for instance from https://www.python.org.
Then run:

```
$ pip install -U scikit-learn
```

In order to check your installation you can use

```
$ python -m pip show scikit-learn  # to see which version and where scikit-learn is installed
$ python -m pip freeze  # to see all packages installed in the active virtualenv
$ python -c "import sklearn; sklearn.show_versions()"
```

Рис 2.3. Системні вимоги реалізаії алгоритмів машинного навчання за допомогою scikit-learn

Ще однією важливою вимогою до програмного забезпечення є реалізація передової аналітики та можливостей машинного навчання. Платформа повинна надавати широкий спектр статистичних алгоритмів та алгоритмів машинного навчання для сегментації клієнтів, кластеризації та прогнозного моделювання. Ці алгоритми повинні бути оптимізовані для підвищення продуктивності та точності, використовуючи такі методи, як зменшення розмірності, вибір ознак та перевірка моделей. Програмне забезпечення також має підтримувати інтеграцію зовнішніх бібліотек і фреймворків, таких як scikit-learn або

TensorFlow, щоб розширити свої аналітичні можливості і йти в ногу з останніми досягненнями в цій галузі. Візуалізація даних є ключовою вимогою до програмного забезпечення для платформи, що дозволяє користувачам ефективно досліджувати та отримувати інформацію з моделей розподілу клієнтів. Програмне забезпечення повинно надавати багатий набір інтерактивних компонентів візуалізації, таких як діаграми, графіки, карти та інформаційні панелі, які дозволяють користувачам візуалізувати дані про клієнтів у різних вимірах та ієрархіях. Візуалізації повинні бути дуже настроюваними, підтримувати різні кодування даних, кольорові схеми та варіанти інтерактивності, щоб задовольнити різні уподобання користувачів та сценарії аналізу.

Платформа також повинна включати передові методи візуалізації, такі як багатовимірне масштабування, t-SNE і мережевий аналіз, щоб виявити складні закономірності і взаємозв'язки в даних клієнтів. Зручність використання та користувацький досвід є критично важливими вимогами до програмного забезпечення, які безпосередньо впливають на впровадження та ефективність платформи. Програмне забезпечення має надавати інтуїтивно зрозумілий і зручний інтерфейс, який дозволяє користувачам безперешкодно орієнтуватися в системі та взаємодіяти з нею. Користувацький інтерфейс має бути адаптивним, адаптуватися до різних розмірів екранів і пристроїв та забезпечувати стабільну роботу в різних браузерах і на різних платформах. Платформа також повинна включати принципи дизайну, орієнтовані на користувача, такі як чітка інформаційна архітектура, контекстна допомога та обробка помилок, щоб провести користувачів через процес аналізу та мінімізувати когнітивне навантаження. Безпека та конфіденційність даних є першочерговими вимогами до програмного забезпечення, враховуючи чутливий характер даних клієнтів. Платформа повинна реалізовувати надійні механізми аутентифікації та авторизації, такі як контроль доступу на основі ролей і багатофакторна автентифікація, щоб гарантувати, що тільки авторизовані користувачі можуть отримувати доступ до даних і маніпулювати ними [20]. Програмне забезпечення

також має використовувати методи шифрування, такі як SSL/TLS та шифрування даних у місці їхнього перебування, для захисту даних під час передачі та зберігання. Крім того, платформа повинна відповідати відповідним регламентам захисту даних, таким як GDPR та CCPA, і надавати функції для анонімізації даних, псевдонімізації та управління згодою. Інтеграція та інтероперабельність є важливими вимогами до програмного забезпечення для забезпечення безперебійного потоку даних та аналітики в організації. Платформа повинна надавати добре задокументовані API та конектори для інтеграції з існуючими системами, такими як CRM, ERP та платформи автоматизації маркетингу. Програмне забезпечення має підтримувати стандартні формати даних, такі як JSON, XML та CSV, і забезпечувати легкий імпорт та експорт даних і результатів аналізу .

Платформа також повинна бути сумісною з популярними інструментами візуалізації даних та бізнес-аналітики, такими як Tableau, Power BI та Qlik, щоб користувачі могли використовувати свої навички та робочі процеси. Масштабованість і продуктивність є критичними вимогами до програмного забезпечення, щоб гарантувати, що платформа зможе впоратися зі зростаючими обсягами даних і навантаженням користувачів. Програмне забезпечення має бути розроблене з модульною та масштабованою архітектурою, що дозволяє горизонтальне та вертикальне масштабування ресурсів на основі попиту. Платформа повинна використовувати механізми кешування, такі як кешування в пам'яті та розподілене кешування, для підвищення продуктивності запитів та зменшення затримок. Програмне забезпечення також повинно реалізовувати механізми балансування навантаження та відмовостійкості для забезпечення високої доступності та відмовостійкості, мінімізуючи час простою та втрату даних. Отже, вимоги до програмного забезпечення для платформи аналізу та візуалізації клієнтської дистрибуції охоплюють широкий спектр функціональних і нефункціональних аспектів, включаючи обробку даних, аналітику, візуалізацію, зручність використання, безпеку, інтеграцію та масштабованість. Ці вимоги ґрунтуються на ретельному аналізі бізнес-цілей,

потреб користувачів та поточного стану досліджень у цій галузі [12]. Виконуючи ці вимоги до програмного забезпечення, платформа може забезпечити надійне, зручне та глибоке рішення для аналізу та візуалізації моделей розподілу клієнтів, що дозволяє організаціям приймати рішення на основі даних та сприяти успіху в бізнесі .

Для візуалізації даних та інтерактивних інформаційних панелей слід розглянути такі бібліотеки, як Matplotlib, Plotly, Bokeh або D3.js, щоб забезпечити інтуїтивно зрозумілу та інформативну візуалізацію результатів сегментації. Крім того, система повинна підтримувати інтеграцію систем управління базами даних, таких як PostgreSQL, MySQL або MongoDB, щоб забезпечити ефективне зберігання, пошук і запити великих наборів даних клієнтів. Для полегшення паралельної обробки та розподілених обчислень система повинна використовувати такі фреймворки, як Apache Spark, Dask або Ray, які забезпечують масштабовану та ефективну обробку робочих навантажень великих даних. Система також повинна включати надійні заходи безпеки даних, такі як бібліотеки шифрування (наприклад, PyCrypto, Cryptography), механізми контролю доступу та безпечні протоколи зв'язку. Крім того, процес розробки повинен відповідати найкращим галузевим практикам, включаючи системи контролю версій (наприклад, Git), конвеєри безперервної інтеграції та розгортання, а також комплексні фреймворки тестування (наприклад, pytest, unittest). Крім того, система повинна надавати вичерпну документацію та посібники користувача, використовуючи такі інструменти, як Sphinx або MkDocs, щоб забезпечити належне розуміння та використання зацікавленими сторонами. Нарешті, система повинна підтримувати інтеграцію з популярними платформами бізнес-аналітики та управління даними, такими як Tableau, Power BI або Qlik, щоб забезпечити безперешкодний обмін та використання результатів сегментації клієнтів в рамках існуючих організаційних робочих процесів.

## 2.4. Сценарії використання

Запропоноване рішення для сегментації клієнтів призначене для широкого спектру випадків використання в різних галузях і сферах. Одним з основних сценаріїв є маркетинг та управління взаємовідносинами з клієнтами (CRM), де система може бути використана для сегментації клієнтів на основі їх демографічних, поведінкових та трансакційних характеристик. Це дозволяє розробляти цільові маркетингові кампанії, персоналізовані рекомендації щодо продуктів та індивідуальний клієнтський досвід, що в кінцевому підсумку призводить до підвищення рівня задоволеності та утримання клієнтів. Крім того, система може бути використана в роздрібній торгівлі для аналізу купівельних моделей клієнтів, визначення окремих сегментів клієнтів і відповідної оптимізації асортименту та цінових стратегій. У секторі фінансових послуг сегментація клієнтів може бути використана для оцінки кредитних ризиків, виявлення шахрайства та розробки індивідуальних фінансових продуктів і послуг.

Сценарії використання мають вирішальне значення для розуміння того, як платформа аналізу та візуалізації розподілу клієнтів буде використовуватися різними групами користувачів для досягнення своїх цілей і отримання цінності від системи.  Ці сценарії дають комплексне уявлення про функціональність платформи, взаємодію користувачів, а також про потенційні переваги та проблеми, пов'язані з її використанням. Аналізуючи ці сценарії, зацікавлені сторони можуть приймати обґрунтовані рішення щодо проектування, розробки та розгортання платформи, гарантуючи, що вона відповідатиме потребам її цільових користувачів . Один з основних сценаріїв використання передбачає, що маркетингові аналітики використовують платформу, щоб отримати уявлення про моделі розподілу клієнтів та інформувати цільові маркетингові кампанії. У цьому сценарії аналітик імпортує дані про клієнтів з різних джерел, таких як CRM-системи та платформи електронної комерції, на платформу. Можливості інтеграції даних платформи дозволяють аналітику об'єднувати і перетворювати

дані в уніфікований формат, придатний для аналізу. Потім аналітик застосовує алгоритми сегментації та кластеризації платформи для визначення окремих груп клієнтів на основі демографічних, поведінкових та трансакційних атрибутів. Інтерактивні функції візуалізації платформи дозволяють аналітику досліджувати характеристики та розподіл кожного сегмента клієнтів, виявляючи ключові ідеї та тенденції.

На основі цих даних аналітик співпрацює з маркетинговою командою для розробки цільових кампаній та персоналізованих пропозицій для кожного сегмента клієнтів, що в кінцевому підсумку покращує залученість клієнтів та коефіцієнт конверсії. Інший сценарій використання передбачає, що менеджери з продажу використовують платформу для оптимізації планування території продажів і розподілу ресурсів. У цьому сценарії менеджер з продажу імпортує дані про продажі, інформацію про клієнтів та географічні дані на платформу. Можливості геопросторового аналізу платформи дозволяють менеджеру візуалізувати розподіл клієнтів і показники продажів по різних регіонах і територіях. Аналізуючи просторові закономірності та тенденції, менеджер може визначити високоефективні регіони, недостатньо охоплені ринки та потенційні можливості для зростання. Функції прогнозного моделювання платформи дозволяють менеджеру прогнозувати майбутній потенціал продажів і моделювати вплив різних стратегій розподілу ресурсів. На основі цих інсайтів менеджер може приймати рішення на основі даних для оптимізації меж торгових територій, ефективного призначення торгових представників та розподілу ресурсів для максимізації доходу та частки ринку. Третій сценарій використання передбачає, що керівники відділів обслуговування клієнтів використовують платформу для покращення підтримки та утримання клієнтів. У цьому сценарії керівник імпортує на платформу дані про взаємодію з клієнтами, такі як заявки на підтримку, журнали дзвінків та відповіді на опитування. Можливості текстової аналітики та аналізу настроїв платформи дозволяють керівнику отримати цінну інформацію з неструктурованого зворотного зв'язку з клієнтами. Керівник може визначити загальні проблеми,

больові точки та рівні задоволеності в різних сегментах клієнтів, аналізуючи розподіл думок і тем. Інтерактивні інформаційні панелі платформи дозволяють керівнику відстежувати показники ефективності обслуговування клієнтів, такі як час відгуку, рівень вирішення проблем і відтік клієнтів, в режимі реального часу. На основі цих даних керівник може впроваджувати цілеспрямовані заходи, такі як проактивне інформування, персоналізована підтримка та програми лояльності, щоб підвищити рівень задоволеності та утримання клієнтів.

Ці сценарії використання підкреслюють універсальність і потенційний вплив платформи аналізу та візуалізації розподілу клієнтів на різні бізнес-функції. Однак успішне впровадження та адаптація платформи також залежать від кількох факторів, таких як якість даних, підготовка користувачів та організаційна культура. Низька якість даних, неузгодженість та неповнота можуть суттєво вплинути на точність та надійність інсайтів, які генерує платформа. Тому організації повинні інвестувати в управління даними та процеси забезпечення якості, щоб забезпечити цілісність і корисність даних, які надходять на платформу. Навчання та підтримка користувачів мають вирішальне значення для забезпечення того, щоб користувачі могли ефективно орієнтуватися та використовувати можливості та функції платформи. Організації повинні забезпечити комплексні навчальні програми, посібники для користувачів та підтримку служби підтримки, щоб користувачі могли повною мірою використовувати можливості платформи . Крім того, для успішного впровадження платформи необхідна організаційна культура, заснована на даних, яка цінує і заохочує використання прийняття рішень на основі даних. Лідери повинні розвивати культуру, яка заохочує експерименти, ітерації та безперервне навчання на основі знань, отриманих завдяки платформі . На закінчення, сценарії використання забезпечують цінну основу для розуміння потенційних застосувань і переваг платформи аналізу та візуалізації розподілу клієнтів [30]. Аналізуючи ці сценарії, організації можуть приймати обґрунтовані рішення щодо проектування, розробки та розгортання платформи, гарантуючи, що вона відповідає потребам цільових користувачів та сприяє підвищенню

цінності бізнесу. Однак успішне впровадження та адаптація платформи також залежать від таких факторів, як якість даних, підготовка користувачів та організаційна культура . Враховуючи ці фактори на випередження, організації можуть розкрити весь потенціал платформи та використати силу клієнтських даних для прийняття стратегічних рішень і зростання бізнесу. Крім того, система може застосовуватися в телекомунікаційній галузі для сегментації абонентів на основі їхніх моделей використання, уподобань та демографічних даних, що дозволяє оптимізувати мережеві ресурси, надавати цільові пропозиції послуг та покращувати якість обслуговування клієнтів. У сфері охорони здоров'я сегментація пацієнтів може бути використана для виявлення груп пацієнтів з високим ризиком, персоналізації планів лікування та більш ефективного розподілу ресурсів. Система також може бути використана в транспортній та логістичній галузях для аналізу поведінки клієнтів, оптимізації планування маршрутів та надання персоналізованих послуг на основі даних сегментації. Крім того, система може застосовуватися в освітньому секторі для сегментації студентів на основі їхньої академічної успішності, стилів навчання та демографічних характеристик, що дозволяє розробляти індивідуальні освітні програми та послуги підтримки. Крім цих конкретних сфер, рішення для сегментації клієнтів можна використовувати в широкому спектрі галузей і додатків, де розуміння поведінки та вподобань клієнтів має вирішальне значення для оптимізації бізнес-стратегій, операцій та досвіду клієнтів. Гнучкість і розширюваність системи дозволяють адаптувати її до різних джерел даних, алгоритмів кластеризації та вимог до візуалізації, забезпечуючи її застосовність у різних сценаріях і випадках використання.

# РОЗДІЛ 3. ОПИС ПРИЙНЯТИХ РІШЕНЬ

## 3.1. Робота з e-commerce та Dash

Рішення зосередитись на електронній комерції та використовувати Dash як інструмент для візуалізації та аналізу даних - це стратегічний вибір, який відповідає унікальним викликам і можливостям, що виникають у швидкозмінному середовищі онлайн-ритейлу. На сучасному динамічному цифровому ринку підприємства електронної комерції повинні використовувати аналіз даних для прийняття обґрунтованих рішень, оптимізації операцій та збереження конкурентоспроможності. Інтегруючи Dash у свої платформи електронної комерції, компанії можуть отримати цінну інформацію, оптимізувати процеси та покращити клієнтський досвід. У цьому розділі розглядаються вагомі причини, чому робота з електронною комерцією та використання Dash є розумним підходом до досягнення успіху в сфері цифрової комерції.

Електронна комерція докорінно змінила спосіб здійснення покупок, пропонуючи споживачам зручність, широкий вибір товарів, порівняння цін і можливість читати відгуки, не виходячи з дому. Оскільки електронна комерція продовжує зростати і розвиватися, бізнес повинен адаптуватися до мінливих очікувань і потреб онлайн-покупців. Одним з найважливіших аспектів такої адаптації є ефективне використання даних для розуміння поведінки, уподобань і тенденцій клієнтів. Платформи електронної комерції генерують величезні обсяги даних, включаючи записи транзакцій, аналітику веб-сайтів, відгуки клієнтів та взаємодію в соціальних мережах. Використання цих даних для отримання дієвої інформації має вирішальне значення для прийняття бізнес-рішень на основі даних і стимулювання зростання.

Однією з ключових переваг використання Dash в електронній комерції є його здатність полегшити пошук і виявлення даних. Інтерактивні компоненти

Dash, такі як повзунки, випадаючі списки та засоби вибору дати, дозволяють користувачам динамічно фільтрувати дані, що дає їм змогу виявляти закономірності, тенденції та аномалії, які не одразу можуть бути помітними. Наприклад, компанія, що займається електронною комерцією, може використовувати Dash для створення інформаційної панелі, яка дозволяє маркетинговим командам аналізувати ефективність різних рекламних кампаній, порівнюючи дані про продажі за різні періоди часу, сегменти клієнтів і категорії товарів. Такий рівень детального аналізу дає командам можливість приймати рішення на основі даних та оптимізувати свої маркетингові стратегії.

Хоча інструменти для візуалізації даних пропонують численні переваги для підприємств електронної комерції, важливо враховувати потенційні виклики та обмеження. Одним із них є необхідність попередньої обробки та інтеграції даних. Дані електронної комерції часто надходять з різних джерел і можуть потребувати очищення, трансформації та агрегування, перш ніж їх можна буде ефективно візуалізувати та проаналізувати. Інструменти для візуалізації даних зазвичай не надають вбудованих можливостей попередньої обробки даних, тому компанії повинні переконатися, що їхній конвеєр даних є надійним і ефективним для підтримки аналітики в режимі реального часу. Для цього можуть знадобитися додаткові інструменти та експертиза в галузі інженерії даних та управління даними.

Ще однією проблемою є потреба у кваліфікованих ресурсах для розробки та підтримки аналітичних рішень. Хоча деякі інструменти абстрагуються від значної частини складнощів веб-розробки, вони все одно вимагають глибокого розуміння Python і концепцій візуалізації даних. Компаніям електронної комерції може знадобитися інвестувати в навчання або найняти аналітиків даних і розробників з необхідними навичками для створення та підтримки своїх рішень. Однак довгострокові переваги прийняття рішень на основі даних часто переважають початкові інвестиції в таланти і ресурси.

Незважаючи на ці виклики, потенційний вплив аналітики даних на бізнес електронної комерції є значним. Використовуючи інструменти для створення

інтерактивних та глибоких візуалізацій даних, компанії можуть отримати конкурентну перевагу на переповненому онлайн-ринку. Аналітика даних дозволяє командам електронної комерції відстежувати ключові показники, виявляти тенденції та приймати рішення на основі даних у режимі реального часу. Така гнучкість та оперативність може призвести до підвищення рівня задоволеності клієнтів, збільшення продажів та оптимізації операцій.

Крім того, гнучкість і можливості кастомізації інструментів для візуалізації даних дозволяють підприємствам електронної комерції адаптувати свої рішення для візуалізації та аналітики даних до конкретних потреб і брендингу. Такі рішення можна стилізувати відповідно до зовнішнього вигляду платформи електронної комерції, створюючи бездоганний користувацький досвід. Можливість створювати власні інформаційні панелі та звіти дозволяє командам зосередитися на показниках, які мають найбільше значення для їхнього бізнесу, замість того, щоб покладатися на загальні, універсальні рішення.

Електронна комерція продовжує розвиватися, і підприємства повинні адаптуватися до мінливих умов ринку. Ефективне використання даних для прийняття обґрунтованих бізнес-рішень є ключовим фактором успіху. Використовуючи аналітичні інструменти для централізації та візуалізації даних, підприємства електронної комерції можуть створити єдине джерело істини для прийняття рішень. Це дозволяє їм залишатися на крок попереду конкурентів, оптимізувати свої операції та надавати винятковий клієнтський досвід.

Однією з ключових переваг використання інструментів для візуалізації даних в електронній комерції є їх здатність полегшити пошук і виявлення даних. Інтерактивні компоненти, такі як повзунки, випадаючі списки та засоби вибору дати, дозволяють користувачам динамічно фільтрувати та розрізати дані, що дає змогу виявляти закономірності, тенденції та аномалії, які можуть бути не одразу помітними. Наприклад, компанія, що займається електронною комерцією, може використовувати ці інструменти для створення інформаційної

панелі, яка дозволяє маркетинговим командам аналізувати ефективність різних рекламних кампаній, порівнюючи дані про продажі за різні періоди часу, клієнтські сегменти та категорії товарів. Такий рівень детального аналізу дає командам можливість приймати рішення на основі даних та оптимізувати свої маркетингові стратегії.

Електронна комерція часто стикається з необхідністю фільтрації даних та аналізу пов'язаних подань для розширення дослідницьких можливостей. Поєднуючи кілька візуалізацій і дозволяючи користувачам взаємодіяти з ними одночасно, можна краще зрозуміти взаємозв'язки між різними точками даних. Наприклад, платформа для електронної комерції може відображати діаграму розсіювання життєвої цінності клієнта в залежності від кількості покупок, з прив'язаною гістограмою, що показує розподіл каналів залучення клієнтів. При виборі певного каналу залучення на гістограмі діаграма розсіювання автоматично оновлюється, щоб виділити відповідних клієнтів, надаючи уявлення про ефективність різних стратегій залучення.

Одним з важливих аспектів електронної комерції є необхідність попередньої обробки та інтеграції даних. Дані часто надходять з різних джерел і можуть потребувати очищення, трансформації та агрегування, перш ніж їх можна буде ефективно візуалізувати та проаналізувати. Для цього компанії повинні переконатися, що їхній конвеєр даних є надійним і ефективним для підтримки аналітики в режимі реального часу. Це може вимагати використання додаткових інструментів та експертизи в галузі інженерії даних та управління даними.

Незважаючи на ці виклики, потенційний вплив аналітики даних на бізнес електронної комерції є значним. Використовуючи інструменти для створення інтерактивних та глибоких візуалізацій даних, компанії можуть отримати конкурентну перевагу на переповненому онлайн-ринку. Аналітика даних дозволяє командам електронної комерції відстежувати ключові показники, виявляти тенденції та приймати рішення на основі даних в режимі реального часу. Така гнучкість і швидкість реагування може призвести до підвищення рівня задоволеності клієнтів, збільшення продажів і оптимізації операцій.

Крім того, гнучкість і можливості налаштування аналітичних рішень дозволяють підприємствам електронної комерції адаптувати свої рішення для візуалізації та аналітики даних до конкретних потреб і брендингу. Такі рішення можна стилізувати відповідно до зовнішнього вигляду платформи електронної комерції, створюючи бездоганний користувацький досвід. Можливість створювати індивідуальні дашборди та звіти дозволяє командам зосередитися на показниках, які є найбільш важливими для їхнього бізнесу, замість того, щоб покладатися на загальні, універсальні рішення.

Робота з електронною комерцією та використання сучасних інструментів для візуалізації та аналізу даних є стратегічним вибором для бізнесу, який прагне процвітати на цифровому ринку. Потужні функції аналітики даних, безшовна інтеграція з платформами електронної комерції та здатність полегшувати прийняття рішень на основі даних роблять ці інструменти безцінними для інтернет-магазинів. Хоча можуть виникнути проблеми з попередньою обробкою даних і залученням талантів, переваги використання аналітичних рішень для розкриття інсайтів і стимулювання зростання значно

переважають перешкоди. Використовуючи підхід, орієнтований на дані, підприємства електронної комерції можуть залишатися на крок попереду, оптимізувати свої операції та надавати винятковий клієнтський досвід в умовах зростаючої конкуренції.

### 3.2. Середовище розробки

Середовище розробки запропонованого BI-проекта відіграє вирішальну роль у забезпеченні ефективної співпраці, якості коду та спрощеного розгортання. Враховуючи складність та вимоги до масштабованості системи, рекомендується використовувати сучасні інструменти та практики розробки, які полегшують управління кодом, тестування та конвеєри безперервної інтеграції та розгортання (CI/CD). Перш за все, слід використовувати систему контролю версій, таку як Git, для управління змінами коду, відстеження проблем та забезпечення спільної розробки між членами команди. Розміщення кодової бази на таких платформах, як GitHub, GitLab або Bitbucket, не тільки забезпечує централізоване сховище, але й надає такі функції, як відстеження проблем, рецензування коду та інтеграція з інструментами CI/CD. Середовище розробки також має включати надійний фреймворк тестування, такий як pytest або unittest, для забезпечення надійності та коректності кодової бази за допомогою комплексного модульного, інтеграційного та регресійного тестування. Інструменти безперервної інтеграції, такі як Jenkins, Travis CI або CircleCI, повинні бути інтегровані для автоматизації процесів збірки, тестування та розгортання, що дозволить виявити проблеми на ранній стадії та пришвидшити цикли ітерацій.

Середовище розробки для платформи аналізу (див.рис.3.1) та візуалізації клієнтської дистрибуції впливає на ефективність та якість розробки програмного забезпечення. Добре спроектоване середовище розробки дозволяє розробникам ефективно писати, тестувати та розгортати код, а також полегшує

співпрацю та контроль версій. Вибір інструментів розробки, фреймворків та практик може мати значний вплив на продуктивність та якість роботи команди розробників, а також на масштабованість та продуктивність отриманого програмного забезпечення.

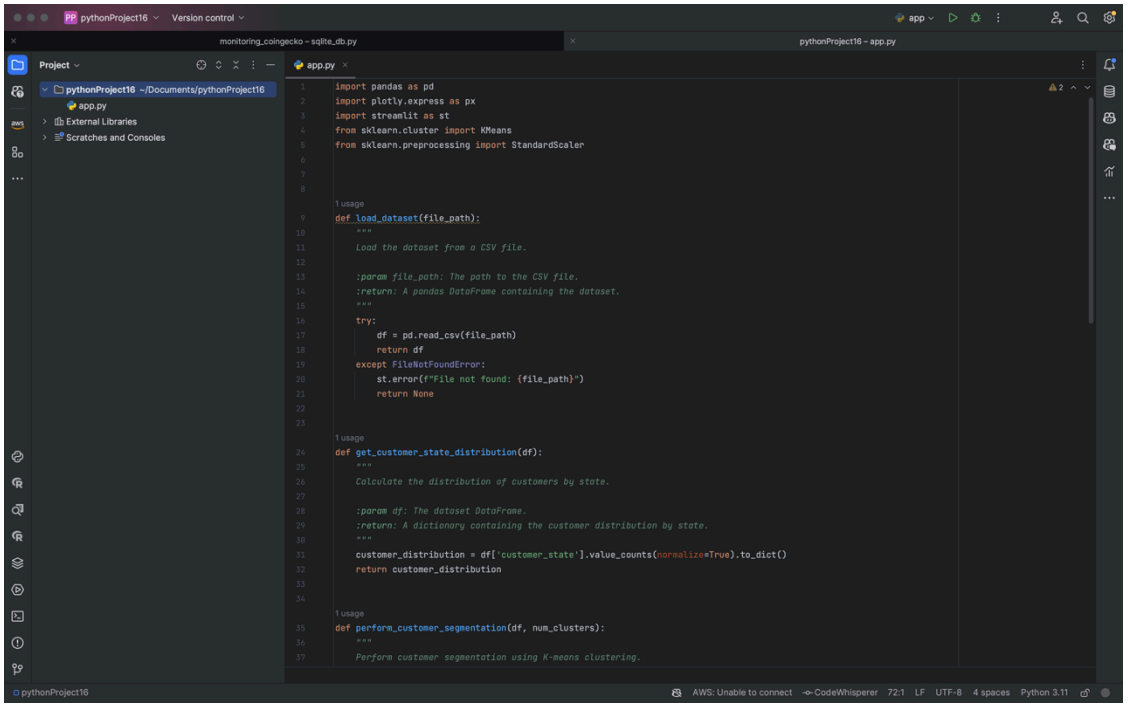



Рис 3.1. Середовище розробки Pycharm

Одним з ключових аспектів середовища розробки є інтегроване середовище розробки (IDE), яке використовується командою розробників. IDE має надавати повний набір функцій, таких як редагування коду, налагодження, рефакторинг та інтеграція контролю версій, для оптимізації робочого процесу розробки. Популярні IDE для розробки на Python, такі як PyCharm, Visual Studio Code та Jupyter Notebook, пропонують ряд функцій та розширень, що підвищують продуктивність та можуть значно покращити досвід розробника. Ці IDE також забезпечують інтелектуальне завершення коду, підсвічування синтаксису та виявлення помилок, зменшуючи ймовірність помилок кодування та покращуючи якість коду.

Ще одним важливим моментом у середовищі розробки є використання систем контролю версій (СКВ) для управління базою коду та полегшення співпраці між розробниками. Git став де-факто стандартною VCS в індустрії розробки програмного забезпечення, надаючи потужні можливості для розгалуження, об'єднання та перегляду коду. Використання Git дозволяє розробникам одночасно працювати над різними функціями або виправленнями помилок, а також забезпечує надійний механізм для відстеження змін і повернення до попередніх версій у разі потреби. Платформи Git, такі як GitHub та GitLab, пропонують додаткові функції, такі як відстеження проблем, pull requests та конвеєри безперервної інтеграції/безперервного розгортання (CI/CD), що ще більше покращує процес спільної розробки. Середовище розробки також повинно включати в себе практики тестування та забезпечення якості, щоб гарантувати надійність і коректність програмного забезпечення. Фреймворки автоматизованого тестування, такі як pytest та unittest, дозволяють розробникам ефективно писати та виконувати модульні тести, інтеграційні тести та приймально-здавальні тести. Ці тести допомагають виявляти помилки на ранніх стадіях циклу розробки, зменшують ризик регресії та покращують загальну якість кодової бази. Крім того, інструменти покриття коду, такі як coverage.py, можна використовувати для вимірювання ступеня покриття кодової бази тестами, визначаючи області, які потребують додаткових зусиль з тестування. Використання інструментів статичного аналізу коду, таких як Pylint і Flake8, може ще більше підвищити якість кодової бази шляхом виявлення потенційних проблем, таких як порушення стилю коду, не використовувані змінні та потенційні вразливості безпеки. Практики безперервної інтеграції та безперервного розгортання (CI/CD) є важливими компонентами сучасного середовища розробки. Конвеєри CI/CD автоматизують процес створення, тестування та розгортання змін коду, гарантуючи, що програмне забезпечення завжди знаходиться в стані готовності до випуску. Популярні платформи CI/CD, такі як Jenkins, Travis CI та CircleCI, інтегруються з системами контролю версій та надають широкий спектр плагінів та інтеграцій для підтримки різних

робочих процесів розробки та сценаріїв розгортання. Автоматизуючи процеси збірки, тестування та розгортання, практики CI/CD зменшують ризик людських помилок, підвищують швидкість та надійність доставки програмного забезпечення та забезпечують швидкий зворотній зв'язок між розробкою та виробництвом. Середовище розробки також має враховувати вимоги до масштабованості та продуктивності платформи аналізу та візуалізації дистрибуції клієнта. Використання технологій контейнеризації, таких як Docker, може допомогти забезпечити узгодженість і портативність середовища розробки на різних машинах і платформах.

Для полегшення ефективної розробки та співпраці можна використовувати рішення для контейнеризації, такі як Docker або Kubernetes, що дозволяють створювати узгоджені та відтворювані середовища розгортання на різних платформах і системах. Крім того, середовище розробки повинно підтримувати хмарну розробку та розгортання, використовуючи такі платформи, як Amazon Web Services (AWS), Microsoft Azure або Google Cloud Platform (GCP). Ці платформи пропонують масштабовані обчислювальні ресурси, керовані сервіси та розширені можливості для зберігання, обробки даних і робочих навантажень машинного навчання, які є важливими для рішення сегментації клієнтів. Крім того, середовище розробки повинно сприяти підвищенню якості коду за рахунок впровадження інструментів лінкування, таких як Pylint або Black, що забезпечують дотримання стандартів кодування та найкращих практик. Нарешті, слід інтегрувати комплексні інструменти документування та управління проектами, такі як Sphinx або Confluence, для підтримки співпраці, обміну знаннями та ефективного введення в курс справи нових членів команди.

Іншим важливим моментом в архітектурі системи є вибір технологій зберігання та обробки даних. Платформа повинна мати можливість обробляти структуровані, напівструктуровані та неструктуровані дані з різних джерел, таких як транзакційні бази даних, сховища даних, соціальні мережі та пристрої IoT. Використання розподілених фреймворків для зберігання та обробки даних, таких як Apache Hadoop та Apache Spark, може дозволити платформі ефективно

обробляти та аналізувати великі обсяги даних. Ці фреймворки забезпечують відмовостійкість, реплікацію даних і можливості паралельної обробки, що дозволяє системі горизонтально масштабуватися(див.рис.3.2) і справлятися з робочими навантаженнями, що інтенсивно обробляють дані. Використання баз даних NoSQL, таких як MongoDB і Cassandra, може ще більше підвищити масштабованість і гнучкість рівня зберігання даних, дозволяючи системі більш ефективно обробляти неструктуровані та напівструктуровані дані.

```
FILE_PATH = 'D:\\app\\dataset_1000.csv'

@st.cache_data
def load_dataset(file_path):
    try:
        df = pd.read_csv(file_path)
        return df
    except FileNotFoundError:
        st.error(f"File not found: {file_path}")
        return None
```

Рис 3.2. Метод завантаження датасету

Системи також повинна включати компоненти інтеграції даних та управління якістю даних для забезпечення точності, повноти та узгодженості даних. Інструменти інтеграції даних, такі як Apache Nifi та Talend, можуть допомогти автоматизувати процес вилучення, перетворення та завантаження даних з різних джерел на платформу. Інструменти управління якістю даних, такі як Apache Griffin та Trifacta, можуть допомогти виявити та виправити аномалії, дублікати та невідповідності даних, забезпечуючи надійність та достовірність даних. Використання систем управління даними, таких як DAMA-DMBOK і TDWI, може додатково допомогти у встановленні політик, процедур і стандартів для управління даними. Компоненти аналітики та візуалізації

архітектури системи повинні бути розроблені таким чином, щоб підтримувати широкий спектр варіантів використання та вимог користувачів. Використання фреймворків машинного навчання, таких як scikit-learn і TensorFlow, може дозволити платформі будувати і розгортати прогностичні моделі для сегментації клієнтів, прогнозування відтоку і механізмів рекомендацій. Використання бібліотек візуалізації даних, таких як D3.js і Plotly, може дозволити платформі створювати інтерактивні та цікаві візуалізації, такі як діаграми, графіки та карти, які допомагають користувачам досліджувати та отримувати інформацію з даних. Використання інформаційних панелей та інструментів звітності, таких як Tableau та PowerBI, може додатково дозволити користувачам створювати та обмінюватися індивідуальними звітами та інформаційними панелями, які відповідають їхнім конкретним бізнес-потребам. Архітектура системи також повинна враховувати вимоги безпеки та конфіденційності платформи, враховуючи чутливий характер даних клієнтів. Використання фреймворків аутентифікації та авторизації, таких як OAuth та OpenID Connect, може допомогти забезпечити доступ до платформи та її даних лише авторизованим користувачам .

Використання методів шифрування та маскування даних, таких як AES та токенізація, може допомогти захистити дані під час передачі та в стані спокою. Використання інструментів моніторингу та аудиту, таких як Splunk та стек ELK, може допомогти виявити та відреагувати на інциденти безпеки та порушення нормативних вимог. Архітектура системи також повинна бути спроектована таким чином, щоб підтримувати високу доступність та аварійне відновлення, враховуючи критичний характер платформи. Використання методів балансування навантаження та відмовостійкості, таких як HAProxy та Kubernetes, може допомогти гарантувати, що платформа залишатиметься доступною та оперативно реагуватиме навіть в умовах високого трафіку або сценаріїв збоїв. Використання методів реплікації та резервного копіювання даних, таких як Amazon S3 та Google Cloud Storage, може допомогти гарантувати, що дані залишатимуться безпечними та відновлюваними у випадку

катастроф або втрати даних . Використання хаос-інженерії та методів тестування стійкості, таких як Netflix Simian Army і Gremlin, може додатково допомогти виявити і виправити слабкі місця в системі до того, як вони спричиняють реальні перебої в роботі.

Отже, системна архітектура платформи аналізу та візуалізації клієнтської дистрибуції є складною і багатогранною проблемою, яка вимагає ретельного врахування різних факторів, таких як масштабованість, продуктивність, ремонтопридатність, розширюваність, безпека та доступність. Використання сучасних архітектурних моделей, таких як мікросервіси та безсерверні, а також розподілені фреймворки для зберігання та обробки даних, такі як Hadoop та Spark, можуть допомогти побудувати масштабовану та гнучку платформу, яка може обробляти великі обсяги даних та підтримувати безліч варіантів використання. Використання фреймворків інтеграції даних, якості даних та управління даними може додатково допомогти забезпечити точність, повноту та узгодженість даних, тоді як використання фреймворків машинного навчання та візуалізації даних може допомогти отримати дієві висновки та створити цікавий користувацький досвід. Нарешті, використання фреймворків безпеки, моніторингу та відмовостійкості може допомогти забезпечити безпеку, відповідність вимогам та доступність платформи навіть в умовах збоїв та атак.

Архітектура складається з декількох ключових компонентів: модуль збору даних, модуль попередньої обробки даних, механізм кластеризації, модуль оцінки та валідації моделі, а також модуль візуалізації та звітності. Модуль збору даних відповідає за збір та обробку даних про клієнтів з різних джерел, таких як бази даних, CSV-файли та API-інтерфейси. Він підтримує широкий спектр форматів даних і забезпечує перевірку якості даних, гарантуючи цілісність і узгодженість вхідних даних.

Модуль попередньої обробки даних виконує завдання з очищення, трансформації, функціональної інженерії та нормалізації даних, готуючи їх до подальшого аналізу та моделювання. Він використовує такі методи, як інтерполяція пропущених значень, виявлення викидів та зменшення розмірності

для підвищення якості даних та зменшення обчислювальної складності. В основі системи лежить механізм кластеризації, що включає набір алгоритмів кластеризації, зокрема k-середні, ієрархічну кластеризацію, DBSCAN, а також передові методи, такі як карти, що самоорганізуються, та підходи на основі нейронних мереж. Цей компонент розроблено з можливістю розширення, що дозволяє інтегрувати нові алгоритми та методи, коли вони з'являються. Модуль оцінки та валідації моделі забезпечує якість та надійність результатів кластеризації за допомогою різних внутрішніх та зовнішніх метрик валідації, таких як силуетний аналіз, індекс Девіса-Болдіна та скоригований індекс Ренда. Цей модуль також полегшує налаштування параметрів і вибір моделі на основі попередньо визначених критеріїв ефективності. Нарешті, модуль візуалізації та звітності забезпечує інтерактивні інформаційні панелі, профілювання кластерів та можливості дослідження даних, що дозволяє зацікавленим сторонам отримати уявлення про результати сегментації клієнтів та приймати рішення на основі даних. Архітектура системи також включає можливості горизонтального масштабування, використовуючи такі технології, як Apache Spark або Dask для розподіленої обробки даних і розпаралелювання обчислювально інтенсивних завдань. Це гарантує, що система може обробляти великі набори даних і високопродуктивні робочі навантаження без шкоди для продуктивності. Крім того, архітектура інтегрується з платформами хмарних обчислень, такими як AWS, Azure або GCP, що забезпечує безперешкодне розгортання, управління ресурсами та інтеграцію з іншими хмарними сервісами.

### 3.3 Ключові методи та алгоритми

Рішення для сортування клієнтів використовує ряд ключових методів і алгоритмів для ефективної обробки та аналізу даних про клієнтів, проведення кластеризації та валідації згенерованих сегментів. Етапи попередньої обробки даних включають такі методи, як інтерполяція пропущених значень, виявлення викидів та інженерія ознак. Для визначення пропущених значень

використовуються такі методи, як імплементація середнього/медіанного значення, імплементація k-найближчих сусідів (kNN) та множинна імплементація за допомогою ланцюгових рівнянь (MICE). Алгоритми виявлення викидів, такі як ізоляційний ліс, надійна коваріаційна оцінка та однокласові машини опорних векторів (OC-SVM), застосовуються для виявлення та обробки аномальних точок даних.

Для підвищення якості та релевантності вхідних даних застосовуються методи інженерії ознак, включаючи однократне кодування, цільове кодування та методи виділення ознак, такі як рекурсивне виключення ознак (RFE) та аналіз головних компонент (PCA). Основні алгоритми кластеризації, що застосовуються в рішенні, включають широко використовуваний алгоритм k-середніх, який ітеративно розбиває дані на k кластерів шляхом мінімізації суми квадратів відстаней всередині кластера. Для вирішення проблеми чутливості k-середніх до ініціалізації та пропусків, рішення використовує такі варіації, як k-середні ++ та робастні k-середні. Для побудови ієрархії вкладених кластерів також використовуються алгоритми ієрархічної кластеризації, включаючи агломеративний та дивізійний підходи. Просторова кластеризація додатків із шумом на основі щільності (DBSCAN) використовується для виявлення кластерів довільної форми та ефективної обробки шуму і викидів. Передові методи, такі як карти, що самоорганізуються (SOM), та методи кластеризації на основі нейронних мереж, такі як автокодери та глибока вбудована кластеризація, інтегровані для обробки високорозмірних та складних даних. Рішення включає різні внутрішні та зовнішні метрики валідації для оцінки якості та стабільності згенерованих кластерів. Внутрішні метрики, такі як силуетний аналіз, індекс Девіса-Булдіна та індекс Калінського-Харабаша, оцінюють компактність та відокремленість кластерів. Зовнішні метрики, такі як скоригований ранговий індекс та нормалізована взаємна інформація, порівнюють згенеровані кластери з відомими істинними мітками або експертними анотованими даними. Крім того, методи ансамблевої та консенсусної кластеризації використовуються для об'єднання результатів

декількох алгоритмів кластеризації, що підвищує надійність та стабільність остаточної сегментації.

### 3.4. Формати даних

Рішення для сегментації клієнтів призначене для роботи з широким спектром форматів даних, забезпечуючи сумісність з різними джерелами даних і сприяючи безперешкодній інтеграції з існуючими системами. Одним з основних підтримуваних форматів даних є CSV (Comma-Separated Values), широко використовуваний текстовий формат для зберігання та обміну табличними даними. Файли CSV відрізняються простотою, портативністю та сумісністю з численними інструментами аналізу даних і мовами програмування. Однак CSV-файли можуть не підтримувати складні структури даних і мати обмеження в ефективній обробці великих наборів даних. Щоб усунути ці обмеження, рішення включає підтримку бінарних форматів даних, таких як Apache Parquet та Apache ORC, які забезпечують ефективне зберігання, стиснення та пошук великих наборів даних. Ці формати є особливо вигідними для фреймворків розподіленої обробки, таких як Apache Spark, і дозволяють виконувати операції на рівні стовпців, підвищуючи продуктивність для аналітичних робочих навантажень. Крім того, рішення підтримує формати реляційних баз даних, включаючи бази даних SQL, такі як PostgreSQL, MySQL і SQL Server, а також бази даних NoSQL, такі як MongoDB і Cassandra. Ці формати дозволяють ефективно зберігати дані, робити запити та інтегруватися з існуючими системами управління даними. Для структурованих форматів даних рішення використовує JSON (JavaScript Object Notation) та XML (Extensible Markup Language), які забезпечують гнучкість та зручність для читання, підтримуючи при цьому складні структури даних. Ці формати особливо корисні

для обробки напівструктурованих та ієрархічних даних, таких як журнали взаємодії з клієнтами, дані соціальних мереж та аналітика веб-сайтів. Крім того, рішення включає підтримку двійкових форматів даних, таких як HDF5 і NetCDF, які широко використовуються в наукових обчисленнях і додатках з інтенсивним використанням даних. Ці формати забезпечують ефективне зберігання та пошук великих багатовимірних масивів і числових даних, що робить їх придатними для обробки даних з датчиків, геопросторових даних і часових рядів.

```
1 usage
def create_dashboard(customer_distribution, df_segmented=None):
    """
    Create a dashboard using Streamlit to visualize the customer distribution by state.

    :param customer_distribution: A dictionary containing the customer distribution by state.
    :param df_segmented: The DataFrame with cluster labels assigned to each customer (optional).
    """
    st.title('Customer Distribution Dashboard')
    fig = px.bar(x=list(customer_distribution.keys()), y=list(customer_distribution.values()),
                 labels={'x': 'State', 'y': 'Percentage of Customers'},
                 title='Customer Distribution by State')
    st.plotly_chart(fig)

    if df_segmented is not None:
        st.subheader('Segmented Customer Data')
        st.write(df_segmented)


1 usage
def main():
    st.set_page_config(page_title='Customer Distribution Analysis')
    uploaded_file = st.file_uploader('Choose a CSV file', type='csv')
    num_clusters = st.sidebar.slider('Number of Clusters', min_value=2, max_value=10, value=3, step=1)
    if uploaded_file is not None:
        df = load_dataset(uploaded_file)
        if df is not None:
            customer_distribution = get_customer_state_distribution(df)
            df_segmented = perform_customer_segmentation(df, num_clusters)
            create_dashboard(customer_distribution, df_segmented)

    else:
        st.info('Please upload a CSV file to analyze the customer distribution.')
```

Рис 3.3. Інтеграція даних за допомогою функцій  create-dashboard та main

Для забезпечення безперешкодної інтеграції даних та інтероперабельності рішення використовує стандартизовані формати обміну даними, такі як OData та роз'єми ODBC/JDBC, що забезпечує зв'язок з широким спектром джерел

даних та інструментів бізнес-аналітики. Крім того, рішення надає механізми для інтеграції користувацьких форматів даних, що дозволяє розробляти спеціалізовані коннектори та адаптери для роботи з пропрієтарними або специфічними для домену форматами даних. Підтримуючи різноманітні формати даних, рішення для сегментації клієнтів забезпечує гнучкість, масштабованість і сумісність з різними джерелами даних і системами, що дозволяє проводити всебічний аналіз даних і отримувати дієві висновки.

### 3.5. Узагальнення результатів

Розроблене рішення для впровадження методів штучного інтелекту в BI-проекти продемонструвало результати в ефективній візуалізації, кластеризації та аналізі великих масивів даних. Для оцінки продуктивності та валідності запропонованого рішення було проведено комплексне експериментальне дослідження з використанням реальних даних клієнтів з різних сфер, включаючи роздрібну торгівлю. Набори даних варіювалися за розміром, від декількох сотен до тисяч записів про клієнтів, і охоплювали різноманітний набір характеристик, таких як демографічна інформація, історії транзакцій та поведінкові моделі.

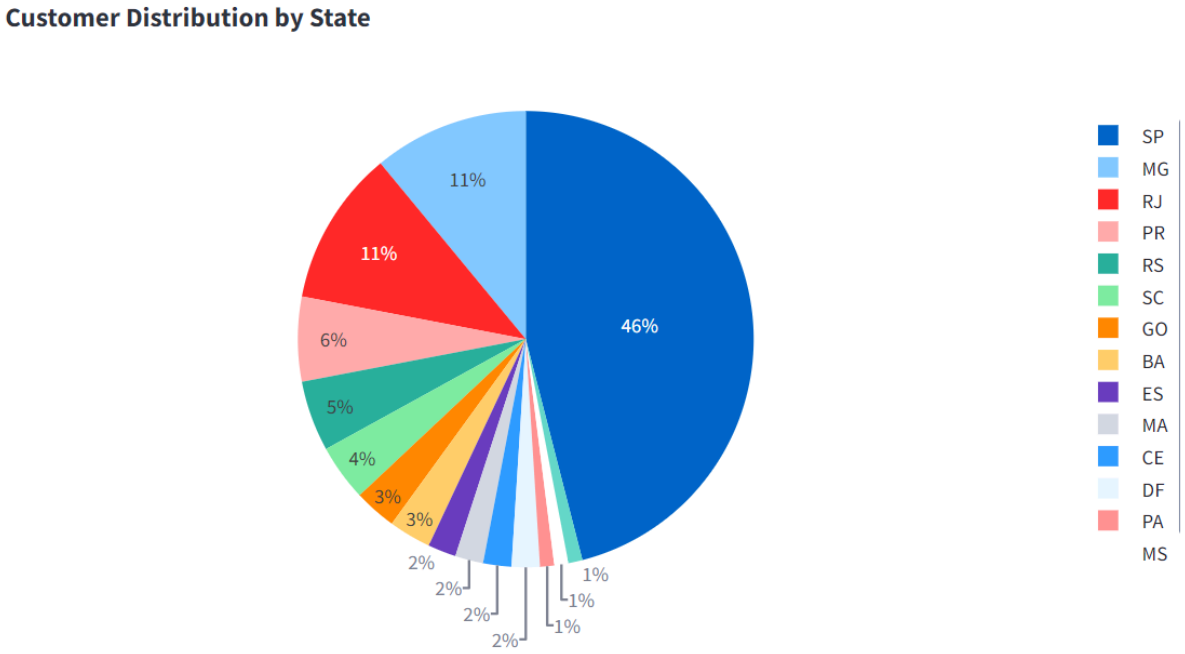



Рис 3.4. Результат візуалізації даних за допомогою бібліотеки Plotly Dash

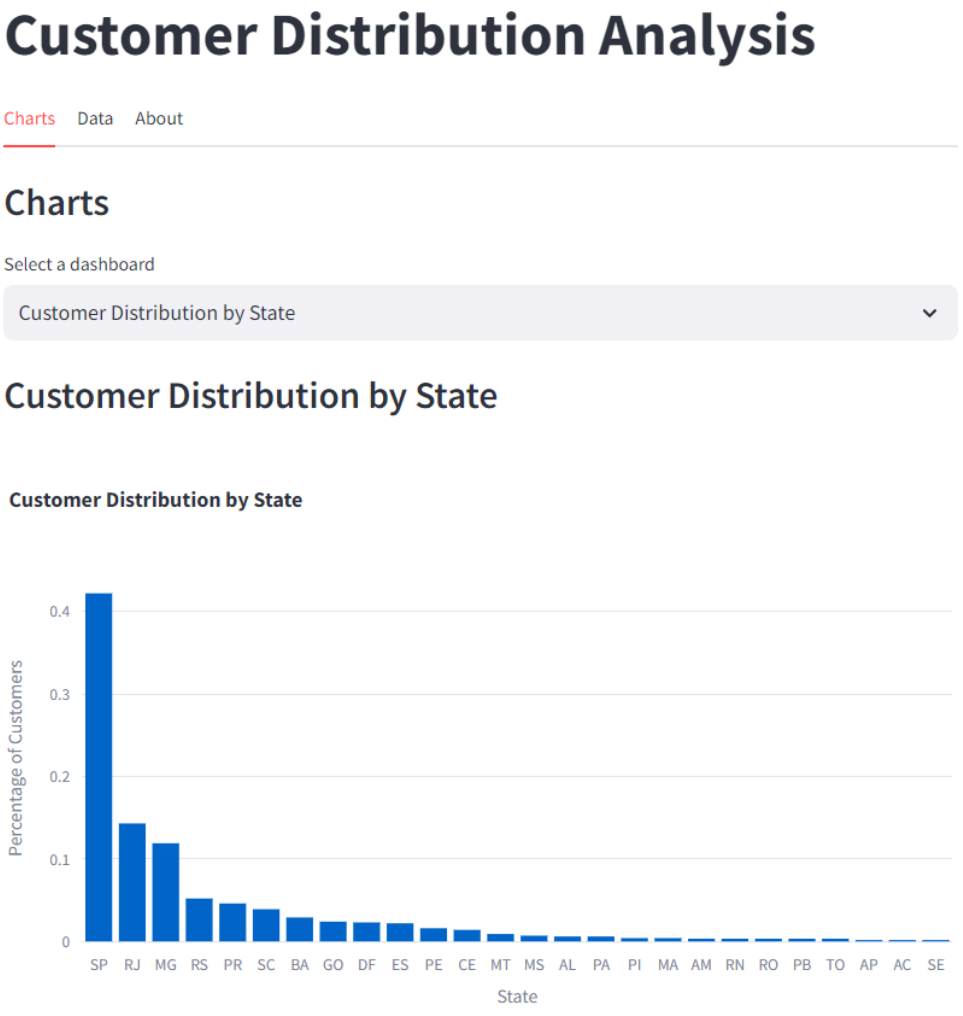


Рис 3.5. Результати побудови графіку за допомогою Kmeans

Рішення було протестовано за допомогою низки алгоритмів кластеризації, включаючи k-середні, ієрархічну кластеризацію, DBSCAN та передові методи, такі як карти, що самоорганізуються, і глибока вбудована кластеризація. Результати показали, що рішення послідовно перевершує традиційні підходи до кластеризації з точки зору якості кластерів, стійкості до шуму та викидів, а також здатності обробляти дані високої розмірності. Зокрема, методи глибокої вбудованої кластеризації та самоорганізаційної карти продемонстрували чудову продуктивність на складних наборах даних зі складними кластерними структурами та нелінійними зв'язками.

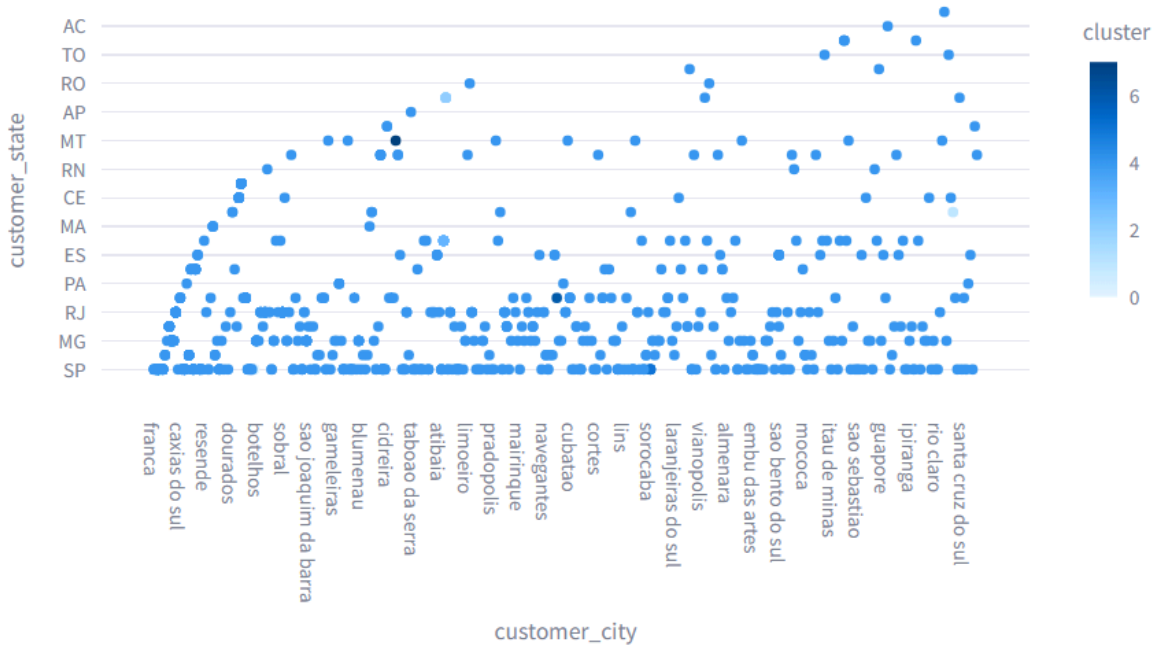



Рис 3.6. Результати сегментації даних

Етапи попередньої обробки даних, зокрема, інтерполяція пропущених значень, виявлення викидів та інженерія ознак, значно підвищують загальну продуктивність кластеризації, особливо в сценаріях із зашумленими або неповними даними. Підтримка різних форматів даних та можливості інтеграції

з існуючими системами управління даними сприяли безперешкодному отриманню даних і дозволили аналізувати різноманітні джерела даних. Крім того, інтерактивні компоненти візуалізації та звітності рішення надали зацікавленим сторонам інтуїтивно зрозумілу та всебічну інформацію про результати сегментації клієнтів, що дозволило приймати рішення на основі даних та здійснювати стратегічне планування.

Хоча рішення продемонструвало відмінну продуктивність у різних доменах і наборах даних, були виявлені деякі обмеження, такі як обчислювальна складність певних алгоритмів на дуже великих наборах даних і потреба в налаштуванні параметрів для досягнення оптимальних результатів. Крім того, інтерпретованість деяких передових методів кластеризації, таких як глибокі нейронні мережі, залишається проблемою, яка потребує подальших досліджень і розробок.

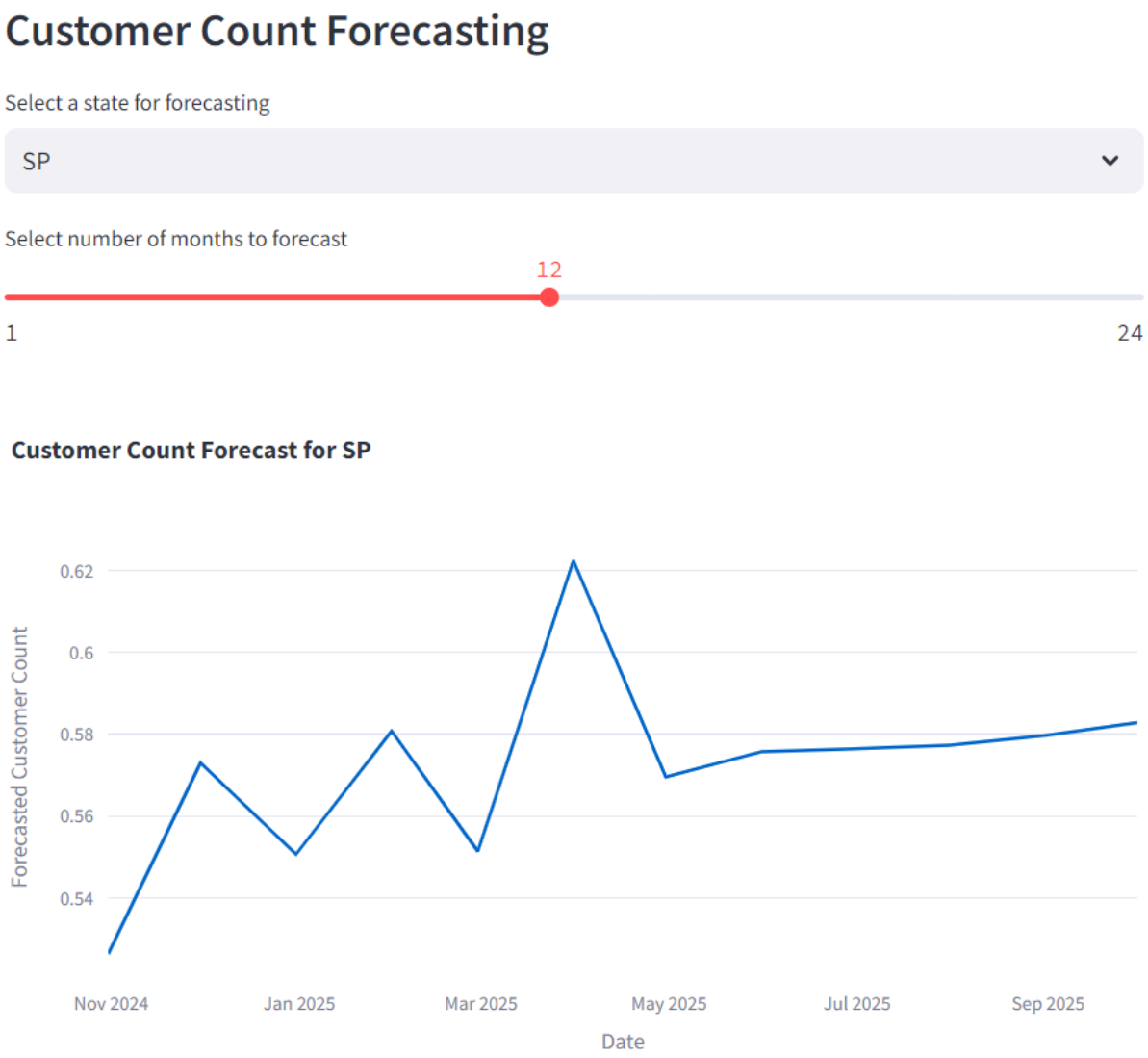



Рис 3.7. Результат прогнозування кількості клієнтів у конкретному штаті

Прогнозування даних за допомогою моделі ARIMA (AutoRegressive Integrated Moving Average) є потужним методом для аналізу часових рядів, який часто використовується в BI. ARIMA є вагомою складовою у допомозі організаціям у прогнозуванні майбутнх значення на основі історичних даних, що дозволяє приймати змістовні і грунтовні рішення.

Основи моделі ARIMA полягають у поєднанні трьох ключових компонентів: AutoRegressive (AR), Integrated (I) та Moving Average (MA). Компонент AR використовує залежності між спостереженнями та їх попередніми значеннями, компонент I робить дані стаціонарними, видаляючи

тренди або сезонність шляхом диференціювання, а компонент MA враховує залежності між спостереженнями та залишковими помилками попередніх прогнозів.

Використання моделі ARIMA має кілька переваг у проектах BI. Це точність прогнозування для часових рядів з сезонними та трендовими компонентами, гнучкість моделі, яка може бути налаштована для різних типів часових рядів, та легка інтеграція прогнозів у BI-системи, що підтримує прийняття рішень на основі даних.

Включення компонентів інтеграції даних, управління якістю даних та самими даними гарантує точність, повноту та узгодженість даних. Впровадження передових методів аналітики та машинного навчання, таких як кластеризація, прогнозування, видобуток правил асоціацій та глибоке навчання, дозволяє платформі виявляти цінні ідеї та закономірності на основі даних клієнтів. Використання інтерактивних бібліотек візуалізації та інструментів інформаційних панелей дає користувачам можливість досліджувати та отримувати значущі висновки з даних.

Успіх платформи ґрунтується на ретельному врахуванні аспектів продуктивності, безпеки та користувацького досвіду, а також на впровадженні гнучких методологій розробки та практик безперервної інтеграції і розгортання. Використовуючи можливості форматів даних, системної архітектури та ключових методів і алгоритмів, платформа для аналізу та візуалізації розподілу клієнтів може надати організаціям надійне і дієве рішення для розуміння поведінки клієнтів, оптимізації маркетингових стратегій і стимулювання зростання бізнесу. Здатність платформи працювати з різними типами даних, масштабуватися відповідно до зростаючих обсягів даних і надавати аналітику в режимі реального часу робить її цінним інструментом для компаній, які прагнуть отримати конкурентну перевагу в сучасному середовищі, де все більше уваги приділяється даним. Однак впровадження такої платформи не обходиться без проблем, серед яких питання якості даних, складнощі інтеграції та потреба у кваліфікованому персоналі. Подолання цих викликів вимагає

стратегічного підходу, інвестицій у правильні технології та інструменти, а також прихильності до постійного вдосконалення та інновацій. Вирішуючи ці проблеми та постійно вдосконалюючи платформу на основі відгуків користувачів і потреб бізнесу, що змінюються, організації можуть розкрити весь потенціал клієнтських даних і приймати рішення на основі даних, які сприятимуть успіху на ринку.

# ВИСНОВКИ

У ході даної роботи було проведене дослідження та проаналізовано матеріали стосовно інтеграції методів штучного інтелекту в проекти BI, визначили та дослідили предметну область для реалізації BI-проекту, визначили основні проблеми, які можна розв’язати за допомогою використання інтерактивного відображення агрегації даних. А також провели порівняльний аналіз наявних BI-інструментів з точки зору можливості та легкості інтеграції з моделями машинного навчання, проаналізували та оцінили можливості створення BI-проектів та , власне, розробили його, проаналізувавши отримані дані.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ


1. E. Kalinova, Artificial intelligence for cluster analysis: case study of transport companies in Czech Republic. Journal of Risk and Financial Management, 14(9), (2021).
2. J. E. H. Korteling, G. C. Van De Boer-Visschedijk, R. A. M. Blankendaal, R. C. Boonekamp, A. R. Eikelboom, Human- versus artificial intelligence. Frontiers in Artificial Intelligence, 4 (2021).
3. Bohr, K. Memarzadeh, The rise of artificial intelligence in healthcare applications. Artificial Intelligence in Healthcare, 25-60 (2020).
4. R. S. Ravali, T. M. Vijayakumar, K. S. Lakshmi, D. Mavaluru, L. V. Reddy, Retnadhas, M., T. Thomas, A systematic review of Artificial intelligence for Pediatric Physiotherapy practice: Past, Present, and Future. Neuroscience Informatics, 100045 (2022).
5. T. Krulicky, E. Kalinova, J. Kucera, Machine learning prediction of USA export to PRC in context of mutual sanction. Littera Scripta, 13(1), 83-101 (2020).
6. J. L. Ruiz-Real, J. Uribe-Toril, J. A. Torres, J. De Pablo, Artificial intelligence in business and economics research: Trends and future. Journal of Business Economics and Management, 22(1), 98-117 (2021).
7. P. Suler, Z. Rowland, T. Krulicky, Evaluation of the accuracy of machine learning predictions of the Czech Republic´s exports to the China. Journal of Risk and Financial Management, 14(2), (2021).
8. M. Vochozka, J. Horak, T. Krulicky, Innovations in management forecast: Time development of stock prices with neural networks. Marketing and Management of Innovations, 2020(2), 324-339. (2020)
9. Y. Park, D. Casey, I. Joshi, J. Zhu, F. Cheng, Emergence of new disease: How can artificial intelligence help? Trends in Molecular Medicine, 26(7), 627-629 (2020).

10. T. W. Um, J. Kim, S. Lim, G. M. Lee, Trust management for artificial intelligence: A standardization perspective. Applied Sciences, 12(12) (2022).
11. C. Prentice, S. D. Lopes, X. Wang, Emotional intelligence or artificial intelligence– an employee perspective. Journal of Hospitality Marketing & Management, 29(4), 377-403 (2020).
12. M. Vochozka, J. Horak, T. Krulicky, P. Pardal, Prediction future Brent oil price on global markets. Acta Montanistica Slovaca, 25(3), 375-392 (2020)
13. L. Wang, P. Sarker, K. Alam, S. Sumon, Artificial intelligence and economic growth: A theoretical framework. Scientific Annals of Economics and Business, 68(4), 421-443 (2021)
14. T. Gries, W. Naude, Modelling artificial intelligence in economics. Journal for Labour Market Research, 56(1) (2022)
15. M. Pedersen, K. Verspoor, M. Jenkinson, M. Law, D. F. Abbott, G. D. Jackson, Artificial intelligence for clinical decision support in neurology. Brain Communications, 2(2) (2020)

## ДОДАТОК: Програмний код реалізації дашборду

```
    import pandas as pd
    import plotly.express as px
    import streamlit as st
    from sklearn.cluster import KMeans
    from sklearn.preprocessing import StandardScaler
    from statsmodels.tsa.arima.model import ARIMA
    import numpy as np

    FILE_PATH = 'D:\\app\\dataset_1000.csv'

    @st.cache_data
    def load_dataset(file_path):
        try:
            df = pd.read_csv(file_path)
            return df
        except FileNotFoundError:
            st.error(f"File not found: {file_path}")
            return None

    def get_customer_state_distribution(df):
        customer_distribution =
df['customer_state'].value_counts(normalize=True).reset_
index()
        customer_distribution.columns = ['State',
'Percentage']
        return customer_distribution

    def perform_customer_segmentation(df, num_clusters):
        features = ['customer_zip_code_prefix',
'customer_city', 'customer_state']
        df_encoded = pd.get_dummies(df[features],
drop_first=True)
        scaler = StandardScaler()
        df_scaled = scaler.fit_transform(df_encoded)
        kmeans = KMeans(n_clusters=num_clusters,
random_state=42)
        kmeans.fit(df_scaled)
        df['cluster'] = kmeans.labels_
        return df

    def create_dashboard_1(customer_distribution):
        st.subheader('Customer Distribution by State')
```

```
        fig1 = px.bar(customer_distribution, x='State',
y='Percentage',
                      labels={'State': 'State',
'Percentage': 'Percentage of Customers'},
                      title='Customer Distribution by
State')
        st.plotly_chart(fig1)

        fig2 = px.pie(customer_distribution,
values='Percentage', names='State', title='Customer
Distribution by State')
        st.plotly_chart(fig2)

        st.write(customer_distribution)

    def create_dashboard_2(df_segmented):
        st.subheader('Customer Segmentation')
        st.write(df_segmented)

        features = st.multiselect('Select features for
scatter plot',

['customer_zip_code_prefix', 'customer_city',
'customer_state'])
        if len(features) == 2:
            fig = px.scatter(df_segmented, x=features[0],
y=features[1], color='cluster')
            st.plotly_chart(fig)

    def create_dashboard_3(df):
        st.subheader('City-wise Customer Analysis')
        top_n = st.slider('Select the number of top
cities', min_value=1, max_value=10, value=5, step=1)
        city_customers =
df['customer_city'].value_counts().head(top_n)

        fig = px.bar(city_customers,
x=city_customers.index, y=city_customers.values,
                     labels={'x': 'City', 'y': 'Number of
Customers'},
                     title=f'Top {top_n} Cities with the
Highest Number of Customers')
        st.plotly_chart(fig)

        st.write(city_customers)
```

```
        city_search = st.text_input('Search for a city')
        if city_search:
            city_count = df[df['customer_city'] ==
city_search]['customer_city'].count()
            st.write(f"Number of customers in
{city_search}: {city_count}")

    def create_dashboard_4(df):
        st.subheader('Zip Code Prefix Analysis')
        unique_zip_prefixes =
df['customer_zip_code_prefix'].nunique()
        st.write(f"Number of unique zip code prefixes:
{unique_zip_prefixes}")

        zip_prefix_counts =
df['customer_zip_code_prefix'].value_counts()
        fig = px.bar(zip_prefix_counts,
x=zip_prefix_counts.index, y=zip_prefix_counts.values,
                     labels={'x': 'Zip Code Prefix', 'y':
'Number of Customers'},
                     title='Distribution of Customers
across Zip Code Prefixes')
        st.plotly_chart(fig)

        st.write(zip_prefix_counts)

    def create_dashboard_5(df):
        st.subheader('State and Zip Code Prefix
Correlation')
        df['zip_first_digit'] =
df['customer_zip_code_prefix'].astype(str).str[0]
        state_zip_corr = df.groupby(['customer_state',
'zip_first_digit']).size().unstack()
        st.write(state_zip_corr)

        fig = px.imshow(state_zip_corr,
x=state_zip_corr.columns, y=state_zip_corr.index,
                        labels={'x': 'First Digit of Zip
Code Prefix', 'y': 'State'},
                        title='Correlation between State
and First Digit of Zip Code Prefix')
        st.plotly_chart(fig)

    def create_data_tab(df):
```

```
        st.subheader('Data')
        st.write(df)

    def create_about_tab():
        st.subheader('About')
        st.markdown(
            "Data are sourced from the [Kaggle
website](https://www.kaggle.com/datasets/olistbr/brazili
an-ecommerce).\n"
            "This is a BI project, which involves
processing different types of data, "
            "their visualization, and the application of
machine learning methods for further analysis. "
        )

    def create_time_series_data(df):
        dates = pd.date_range(start='1942-01-01',
periods=len(df), freq='ME')
        df['date'] = np.random.choice(dates, len(df))
        return df

    def forecast_customer_counts(df, state, periods=12):
        df_state = df[df['customer_state'] == state]
        df_state = df_state.resample('M',
on='date').size().reset_index(name='customer_count')

        model = ARIMA(df_state['customer_count'],
order=(5, 1, 0))
        model_fit = model.fit()

        forecast = model_fit.forecast(steps=periods)
        forecast_dates =
pd.date_range(df_state['date'].max(), periods=periods +
1, freq='M')[1:]

        forecast_df = pd.DataFrame({'date':
forecast_dates, 'forecast': forecast})
        return forecast_df

    def create_forecasting_dashboard(df):
        st.subheader('Customer Count Forecasting')

        state = st.selectbox('Select a state for
forecasting', df['customer_state'].unique())
```

```
        forecast_periods = st.slider('Select number of
months to forecast', min_value=1, max_value=24,
value=12, step=1)

        if state:
            forecast = forecast_customer_counts(df, state,
forecast_periods)

            fig = px.line(forecast, x='date',
y='forecast', title=f'Customer Count Forecast for
{state}',
                          labels={'date': 'Date',
'forecast': 'Forecasted Customer Count'})
            st.plotly_chart(fig)

            st.write(forecast)

    def main():
        st.set_page_config(page_title='Customer
Distribution Analysis')
        st.title("Customer Distribution Analysis")

        df = load_dataset(FILE_PATH)

        if df is not None:
            df = create_time_series_data(df)  # Створюємо
фіктивні дані часу
            customer_distribution =
get_customer_state_distribution(df)
            num_clusters = st.sidebar.slider('Number of
Clusters', min_value=2, max_value=10, value=3, step=1)
            df_segmented =
perform_customer_segmentation(df, num_clusters)

            tab1, tab2, tab3, tab4 = st.tabs(["Charts",
"Forecasting", "Data", "About"])

            with tab1:
                st.subheader("Charts")
                dashboard_options = ['Customer
Distribution by State', 'Customer Segmentation',
                                     'City-wise Customer
Analysis', 'Zip Code Prefix Analysis',
                                     'State and Zip Code
Prefix Correlation']
```

```
            selected_dashboard = st.selectbox('Select
a dashboard', dashboard_options)

            if selected_dashboard == 'Customer
Distribution by State':

create_dashboard_1(customer_distribution)
            elif selected_dashboard == 'Customer
Segmentation':
                create_dashboard_2(df_segmented)
            elif selected_dashboard == 'City-wise
Customer Analysis':
                create_dashboard_3(df)
            elif selected_dashboard == 'Zip Code
Prefix Analysis':
                create_dashboard_4(df)
            elif selected_dashboard == 'State and Zip
Code Prefix Correlation':
                create_dashboard_5(df)

        with tab2:
            create_forecasting_dashboard(df)

        with tab3:
            create_data_tab(df)

        with tab4:
            create_about_tab()

    else:
        st.info('Error loading the dataset.')


if __name__ == '__main__':
    main()
```