НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

На правах рукопису
УДК 004.8

До захисту допущено
Завідувач кафедри ММСА
_____ Оксана ТИМОЩУК
«___» _____ 2023 р.

Магістерська дисертація
на здобуття ступеня магістра
за освітньо-професійною програмою «Системний аналіз фінансового ринку»
зі спеціальності 124 «Системний аналіз»
на тему: «Система підтримки прийняття рішень для оптимізації рекламних кампаній підприємства на основі методу моделювання впливу з залежним представленням даних»

Виконав:
Студент 2 курсу, групи КА-22мп
Заіка Богдан Юрійович _____

Науковий керівник: професор кафедри ММСА,
д.т.н., доц. Терентьєв Олександр Миколайович _____

Рецензент: с.н.с. відділу
прикладної інформатики Інституту
телекомунікацій і глобального
інформаційного простору
к.е.н., доц. Тетяна Іванівна Просянкіна-Жарова _____

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань
Студент (*підпис*): _____

Київ - 2023

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

Рівень вищої освіти — другий (магістерський)
Спеціальність — 124 «Системний аналіз»
Освітньо-професійною програмою «Системний аналіз фінансового ринку»

ЗАТВЕРДЖУЮ
Завідувач кафедри ММСА
_____ Оксана ТИМОЩУК
«___» _____ 2023 р.

ЗАВДАННЯ
на магістерську дисертацію студенту
Заіці Богдану Юрійовичу

1. Тема дисертації: «Система підтримки прийняття рішень для оптимізації рекламних кампаній підприємства на основі методу моделювання впливу з залежним представленням даних», науковий керівник дисертації д.т.н., доц. Терентьєв Олександр Миколайович, затверджені наказом по університету від «08» листопада 2023 р. № 5200-с

2. Строк подання студентом дисертації: _____

3. Об'єкт дослідження: дані про учасників рекламної кампанії підприємства.

4. Предмет дослідження: методи моделювання впливу взаємодії з клієнтами на виконання ними цільової дії та критерії оцінки якості створених моделей впливу.

5. Перелік завдань, які потрібно розробити:
1) дослідити актуальність обраної теми;
2) провести аналіз теоретичного матеріалу щодо методів моделювання впливу та оцінки якості створених моделей;
3) провести порівняльний аналіз обраних методів моделювання впливу;
4) розробити систему підтримки прийняття рішення на основі обраних методів моделювання впливу та критеріїв оцінки якості створених моделей;
5) розробити план стартап-проєкту за темою роботи.

6. Перелік графічного (ілюстративного) матеріалу:
1) графіки та таблиці метрик оцінки якості натренованих моделей впливу;
2) відображення реалізації СППР у вигляді блок-схеми та UML-діаграми;
3) знімки інтерфейсу реалізованої СППР;
4) таблиці, пов’язані з аналізом стартап-проєкту.

7. Орієнтовний перелік публікацій: Заіка Б. Ю., Терентьєв О. М. Система підтримки прийняття рішень для оптимізації рекламних кампаній підприємства на основі методу моделювання впливу з залежним представленням даних, ІІ науково-практична конференція «Системні науки та інформатика», КПІ ім. Ігоря Сікорського, Київ, 4-8 грудня, 2023. С. 108-116.

8. Дата видачі завдання: 1 вересня 2023 року

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1	Формулювання теми магістерської дисертації.	01.09.2023 – 08.09.2023	Виконано
2	Огляд літературно-інформаційних джерел за темою роботи.	09.09.2023 – 22.09.2023	Виконано
3	Виділення моделей і критеріїв для дослідження.	23.09.2023 – 30.09.2023	Виконано
4	Порівняльний аналіз та проектування архітектури програмного продукту.	01.10.2023 – 15.10.2023	Виконано
5	Програмна реалізація створеної архітектури.	16.10.2023 – 29.10.2023	Виконано
6	Розробка інтерфейсу.	30.10.2023 – 06.11.2023	Виконано
7	Опис стартап-проєкту.	07.11.2023 – 13.11.2023	Виконано
8	Оформлення пояснювальної записки.	14.11.2023 – 25.12.2023	Виконано

Студент _____ Богдан ЗАІКА

Науковий керівник дисертації _____ Олександр ТЕРЕНТЬЄВ

РЕФЕРАТ

Магістерська дисертація: 93 с., 32 рис., 24 табл., 1 додаток, 22 джерела.

Ключові слова: МОДЕЛЮВАННЯ ВПЛИВУ, СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ, РЕКЛАМНА КАМПАНІЯ, КЕРОВАНЕ НАВЧАННЯ.

Актуальність роботи зумовлена необхідністю для підприємства оперативного формування списків потенційно прибуткових клієнтів для комунікації на основі даних про учасників попередніх схожих рекламних кампаній. Основною перевагою розглянутих методів моделювання впливу є використання даних і тестової, і контрольної груп на відміну від традиційних методів моделювання.

Об’єктом дослідження є дані про учасників рекламної кампанії підприємства.

Предметом дослідження є методи моделювання впливу взаємодії з клієнтами на виконання ними цільової дії та критерії оцінки якості створених моделей впливу.

Метою роботи є порівняльний аналіз методу моделювання впливу з залежним представленням даних з іншими методами моделювання впливу та формування на їх основі системи підтримки прийняття рішення, яка дозволяє проаналізувати дані, натренувати на їх основі моделі впливу, порівняти їх та використати найкращу на новому наборі даних.

Результатом роботи є створення системи підтримки прийняття рішень для моделювання впливу рекламних кампаній на користувачів. Програмний продукт реалізовано на мові програмування Python з використанням бібліотеки Streamlit для формування інтерфейсу.

ABSTRACT

Master's Thesis: 93 p., 32 fig., 24 tabl., 1 appendix, 22 ref.

Keywords: UPLIFT MODELING, DECISION SUPPORT SYSTEM, ADVERTISING CAMPAIGN, SUPERVISED LEARNING.

The relevance of the work is determined by the need for the enterprise to quickly form lists of potentially profitable clients for communication based on data about participants of previous similar advertising campaigns. The main advantage of the considered uplift modeling methods is the use of both test and control groups data in comparison with traditional modeling methods.

The object of the research is data on the participants of the company's advertising campaign.

The subject of the research is the set of methods of modeling the uplift of interaction with clients on their performance of the target action and criteria for evaluating the quality of the created uplift models.

The purpose of the work is a comparative analysis of the uplift modeling method with dependent data presentation with other uplift modeling methods and the formation of a decision support system based on them, which allows analyzing data, training uplift models based on it, comparing them and using the best one on a new data set.

The result of the work is the creation of a decision support system for uplift modeling of advertising campaigns on users. The software product is implemented in the Python programming language using the Streamlit library to create the interface.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ

СППР – Система підтримки прийняття рішень

DDR – Dependent data representation

T – Test group

C – Control group

ВСТУП

У сучасному світі динамічного розвитку технологій та конкурентної боротьби у сфері бізнесу, прогресивні методи аналізу даних стають крайньою необхідністю. Рекламні кампанії є невід'ємною складовою будь-якого підприємства та потребують правильного підходу для раціонального використання ресурсів та уникнення негативного впливу на клієнтів. Проблемою традиційних підходів моделювання в даному контексті є фокусування на прогнозі ймовірності виконання цільової дії користувачем після комунікації з ним. Моделювання впливу, на відміну від традиційних моделей, прогнозує вплив взаємодії з користувачем на ймовірність виконання ним цільової дії. Використання такого підходу дозволяє раціональніше використовувати ресурси компанії, фокусуючи комунікацію на користувачів, взаємодія з якими матиме найбільший позитивний вплив на виконання ними цільової дії.

У магістерській дисертації розглядаються методи моделювання впливу для вирішення вищезгаданої задачі, критерії оцінки якості створених моделей впливу та реалізація системи підтримки прийняття рішень для можливості швидко тренувати та використовувати створені моделі впливу для нових рекламних кампаній.

Завданнями даної роботи є:

1) дослідити актуальність обраної теми;

2) провести аналіз теоретичного матеріалу щодо методів моделювання впливу та оцінки якості створених моделей;

3) провести порівняльний аналіз обраних методів моделювання впливу;

4) розробити систему підтримки прийняття рішення на основі обраних методів моделювання впливу та критеріїв оцінки якості створених моделей;

5) розробити план стартап-проєкту за темою роботи.

До складу роботи входять вступ, чотири розділи, висновки, перелік джерел посилання та додаток.

В першому вступному розділі обґрунтована актуальність теми дослідження, проаналізовано попередні дослідження даної тематики, а також визначено предмет і об’єкт дослідження, задачі, які будуть вирішені в процесі написання роботи.

В другому розділі проведено аналіз теоретичного матеріалу по моделюванню впливу. Розглянуто методи моделювання впливу та метрики оцінки якості створених моделей впливу, які реалізовано та проаналізовано в розділі 3.

В розділі 3 проведено порівняльний аналіз обраних методів моделювання впливу та реалізовано повноцінну та інтуїтивно зрозумілу СППР на їх основі. Сформульовано алгоритм роботи СППР, надано опис основних класів, інтерфейсу і прикладів використання.

В розділі 4 розроблено стартап проєкт за темою дисертації, проаналізовано його ринкові можливості та створено план виведення його на ринок.

Перелік джерел посилання містить 22 джерела.

В рамках роботи використовувались:

- веб-браузер Google Chrome для пошуку інформації;

- текстовий редактор Microsoft Word для оформлення магістерської дисертації;

- мова програмування Python 3.10, середовище розробки Visual Studio Code та бібліотека Streamlit для розробки СППР.

РОЗДІЛ 1 АНАЛІЗ І ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Актуальність проблеми дослідження

У сучасному світі великої конкуренції та постійних змін, використання даних для розуміння поведінки споживачів та ефективного впливу на них стали критичними завданнями для бізнесу. Рекламні кампанії мають велике значення для підприємства, оскільки вони забезпечують залучення уваги клієнтів, збільшення продажів, підтримку конкурентоспроможності та комунікацію з аудиторією. Реклама дозволяє підтримувати зв'язок зі споживачами, повідомляти їх про новинки, акції, зміни та відповідати на їхні запитання. Проте нераціональне використання цього інструменту бізнесом може завдавати збитків у вигляді ресурсів та клієнтів, через зайві комунікації або невдалу персоналізацію реклами.

Традиційні методи класифікації прогнозують ймовірність належності користувача до класу користувачів, які зробили цільову дію після взаємодії з ними. На основі цих прогнозів часто приймаються рішення щодо взаємодії з класифікованими особами. Однак справжньою метою рекламних кампаній має бути визначення різниці у виконанні цільової дії користувачем з та без взаємодії з ним. Традиційні методи класифікації не використовують інформацію стосовно контрольних груп і тому мають обмежене застосування в цьому контексті.

На відміну від них, моделювання впливу дозволяє включати контрольну групу та спрямовується на явне моделювання різниці в ймовірності результату між двома групами, тому воно набагато краще підходить для аналізу потенційних отримувачів реклами. Крім того, моделі впливу дозволяють безпосередньо ідентифікувати користувачів, взаємодія з якими є найефективнішою [1]. Такий підхід до комунікації з клієнтами дозволяє

раціонально використовувати ресурси та уникати контакту з клієнтами, яких реклама може відштовхнути від виконання цільової дії. Саме в цьому і полягає актуальність використання моделювання впливу для прийняття рішень стосовно рекламних кампаній підприємства: можливість спрогнозувати як взаємодія вплине на користувача, щоб уникнути небажаної комунікації та розуміти з якими користувачами вигідніше взаємодіяти для досягнення цілі.

1.2 Огляд попередніх досліджень та результатів використання моделювання впливу

Інтерес авторів до моделювання впливу виник у близько 1996 року під час консультування та створення комерційного програмного забезпечення для аналітичного маркетингу. На той час найбільш поширеними методами моделювання для спрямованості були різні форми регресії та дерева. Зазвичай до більш популярних методів регресії входили лінійна регресія, логістична регресія та загальні адитивні моделі [2], як правило, у формі рейтингових карток. До популярних методів на основі дерев входили класифікаційні та регресійні дерева. За допомогою цих методів будували моделі, що визначали чи виконає клієнт цільову дію після взаємодії з ним. Швидко стало зрозуміло, що вони не приводили до оптимального розподілу ресурсів прямого маркетингу з вищезгаданих причин. Як наслідок, вони не дозволяли точно спрямовувати зусилля на тих людей, які найбільше позитивно реагували на маркетинговий вплив.

Моделі впливу мають на меті вирішити це завдання, відрізняючи два типи клієнтів з позитивним відгуком: тих, хто мав схильність реагувати позитивно навіть без маркетингового втручання (лояльні користувачі), і тих, хто позитивно реагував на маркетингову дію (схильні до переконання

користувачі). Таким чином, моделі впливу визначають осіб, які не були схильні до виконання цільової дії, поки на них не вплинула маркетингова кампанія.

Ця концепція здобула популярність через свій потенціал суттєво поліпшити розподіл маркетингових ресурсів, акцентуючи увагу на особах, які найбільше схильні до позитивного впливу маркетингових кампаній, тим самим максимізуючи видачу інвестицій.

З часом було розроблено різноманітні підходи та алгоритми для впровадження моделей впливу, використовуючи техніки машинного навчання, статистики та економетрії для ефективнішого визначення та спрямування на схильних до переконання осіб.

З моменту свого виникнення моделі впливу знайшли застосування не лише у маркетингу, але й у галузях, таких як охорона здоров'я, соціальні науки та інші, де метою є розуміння причинної залежності дій на поведінку окремих осіб. Постійна еволюція та удосконалення методик моделей впливу робить їх все більш цінним інструментом для оптимізації спрямованих дій та розподілу ресурсів у різних сферах [3-5].

Впровадження в процеси моделювання впливу неодноразово показувало успішні результати.

1. Банк США виявив, що модель відгуку була дуже неефективною для спрямування (фізичної) розсилки, просуваючи високоякісний продукт своїм існуючим клієнтам. Коли комунікація була спрямована на всю базу, це було прибутково (з урахуванням різниці об’ємів продажів порівняно з контрольною групою). Але коли націлювались на топ 30% користувачів, визначених звичайною моделлю "відгуку", результат був майже нульовим у прирості продаж (і внаслідок цього від'ємний показник ROI). Причиною було те, що модель "відгуку" спрямовувала зусилля на людей, які все одно виконали б цільову дію. Модель впливу змогла ідентифікувати інший топ 30% користувачів, комунікація з якими забезпечила 90% приросту в продажах, порівнюючи комунікацію з усією базою користувачів, і відповідно перетворила невдалий маркетинговий захід у високоефективний і прибутковий [6].

2. Ініціатива зі зменшення відтоку користувачів в компанії мобільного зв'язку фактично збільшила відтік з 9% до 10%. Після цього було створено модель впливу та використано для визначення сегменту у розмірі 30% користувачів, спрямовуючи увагу лише на який, вдалося знизити загальний відтік з 9% до менше 8%, при цьому скоротивши витрати на 70% [7].

3. Ініціатива у іншого мобільного оператора зі зменшення відтоку користувачів була успішною у зменшенні відтоку приблизно на 5%, але модель впливу змогла ідентифікувати 25% користувачів, де ініціатива не приносила прибутку або навіть завдавала збитків. Спрямовуючи зусилля лише на ідентифікованих 75% всього відтоку, загальне утримання користувачів зросло з 5% до 6% (вдалося зберегти на 20% більше клієнтів за менші витрати) [7].

1.3 Висновки до розділу 1 та постановка задачі дослідження

Даний розділ дисертації був присвячений висвітленню актуальності проблеми визначення найвигідніших користувачів для комунікації в рамках рекламної кампанії та переваг моделювання впливу для вирішення цієї проблеми в порівнянні з традиційними методами моделювання. Розглянуто декілька прикладів, де моделювання впливу допомогло краще оптимізувати використання ресурсів.

Постановка задачі дослідження в рамках даної роботи полягає у порівняльному аналізі методу DDR моделювання впливу з іншими методами та реалізації на їх основі повноцінної та інтуїтивно зрозумілої СППР, що стане невід’ємним інструментом аналітиків та бізнес користувачів підприємства для швидкого аналізу та створення моделей впливу для проведення оперативних та персоналізованих рекламних кампаній.

РОЗДІЛ 2 ОБРАНІ МЕТОДИ І ПІДХОДИ ДО МОДЕЛЮВАННЯ ВПЛИВУ

2.1 Постановка задачі моделювання впливу в контексті рекламної кампанії

Нехай набір даних містить інформацію про N учасників рекламної кампанії, в рамках якої учасники випадковим чином були поділені на тестову T та контрольну C підгрупи (розмір підгруп N^T та N^C відповідно). З підгрупою T проведено комунікацію, а з підгрупою C – ні. Виконання цільової дії користувачами позначається як вектор $Y = (y_i | i = \overline{1,N}, y_i \in \{0,1\})$, де $y_i = 1$, якщо i-ий користувач виконав цільову дію, інакше 0.

В звичайних методах модель навчають передбачати ймовірність, що $y_i = 1$ за умови комунікації з i-им користувачем. Тому тренування моделі проводиться тільки на даних про підгрупу T, а інформація про підгрупу C не використовується.

На відміну від традиційних, методи моделювання впливу мають за мету навчити модель передбачати різницю в ймовірностях, що $y_i = 1$ в залежності від наявності комунікації з i-им користувачем. Тому під час тренування використовуються дані про обидві підгрупи T та C.

Отже, звичайна модель намагається оцінити ймовірність того, що клієнти зроблять цільову дію, якщо ми проведемо комунікацію з ними, а модель впливу намагається оцінити збільшення їх ймовірності покупки, якщо ми проведемо комунікацію з ними, в порівнянні з відповідною ймовірністю, якщо ми цього не робимо. Явна мета полягає в тому, щоб моделювати різницю у поведінці відносно виконання цільової дії між підгрупами T і C [3].

2.2 Види користувачів в моделюванні впливу

В моделюванні впливу прийнято ділити користувачів на чотири групи.

1. Схильні до переконання (Persuadables) – користувачі, які виконають цільову дію після комунікації з ними, але без комунікації не виконали б. Це цільова група, яку необхідно визначити моделлю впливу для найприбутковішої комунікації.

2. Лояльні (Sure Things) – користувачі, які виконають цільову дію незалежно від наявності комунікації з ними. Взаємодія з цією групою користувачів не принесе додатково доходу, проте створить додаткові витрати.

3. Втрачені (Lost Causes) – користувачі, які ніколи не виконають цільову дію незалежно від наявності комунікації з ними. Аналогічно лояльним, комунікація з такими користувачами тільки несе за собою зайві витрати.

4. Сплячі собаки (Sleeping Dogs) – користувачі, які без комунікації виконають цільову дію, але у випадку комунікації з ними негативно відреагують та не виконають цільову дію. Це найгірша група для комунікації, оскільки комунікація з ними одночасно збільшує витрати та зменшує дохід [8].

Така класифікація користувачів показує перевагу моделювання впливу в порівнянні з звичайним моделюванням. Звичайна модель не вміє розрізняти схильних до переконання та лояльних користувачів і оскільки лояльні користувачі зазвичай найактивніші, то ймовірно звичайна модель буде оцінювати їх вище за рівнем відгуку (і відповідно за пріоритетом комунікації) ніж схильних до переконання, як це було у вищезгаданій ситуації в банку США. Натомість моделі впливу пріоритезують користувачів, комунікація з якими найбільше збільшить ймовірність виконання ними цільової дії, в порівнянні з ситуацією без комунікації.

2.3 Моделі впливу

В даній роботі розглядаються три представники моделей впливу виду мета-навчання (Meta-learners), які використовують звичайні ймовірнісні моделі класифікації з певним принципом перетворення вхідних або вихідних даних для створення моделей впливу: S-Learner, Z-Learner та Dependent Data Representation (DDR).

2.3.1 S-Learner

Ідея моделі впливу S-Learner полягає в подачі ознаки наявності комунікації з користувачем як додаткової ознаки на вхід моделі. Таким чином під час тренування модель вивчає вплив наявності комунікації на виконання цільової дії клієнтом.

Нехай $X_{train} \in \mathbb{R}^{n \times k}$ – значення k ознак, що описують n користувачів в тренувальній вибірці; $W_{train} \in \mathbb{R}^{n \times 1}$ – ознака проведення комунікації з користувачами в тренувальній вибірці, де $w_i = 1$ означає що проводилась комунікація з i-им користувачем, інакше 0; $Y_{train} \in \mathbb{R}^{n \times 1}$ – ознака виконання цільової дії клієнтами в тренувальній вибірці, де $y_i = 1$ означає що i-ий користувач виконав цільову дію, інакше 0. Тоді процес тренування зображено на рисунку 2.1.

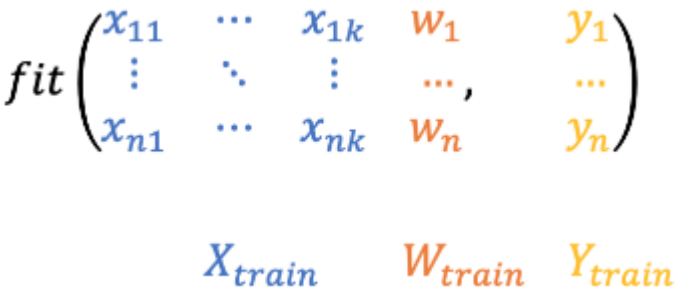


Рисунок 2.1. Процес тренування S-Learner моделі

Далі для оцінки впливу комунікації на m нових користувачів, значення їх ознак передаються двічі в отриману модель: з $W_1 = (1|i = \overline{1,m})$, щоб оцінити ймовірність позитивного відгуку користувачів при комунікації з ними, та з $W_0 = (0|i = \overline{1,m})$, щоб оцінити ймовірність позитивного відгуку при її відсутності. Різниця отриманих значень є шуканою оцінкою впливу комунікації на користувачів. Нехай $X_{test} \in \mathbb{R}^{m \times k}$ – значення k ознак, що описують m нових користувачів; $uplift \in \mathbb{R}^{m \times 1}$ – оцінка моделі впливу комунікації на m нових користувачів. Тоді формула використання натренованої моделі для оцінки впливу комунікації m нових користувачів зображена на рисунку 2.2 [5, 9].

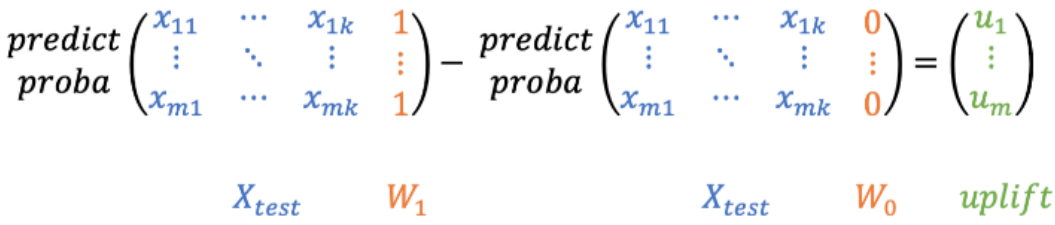


Рисунок 2.2. Формула використання S-Learner моделі для оцінки впливу комунікації

2.3.2 Z-Learner

Ідея моделі впливу Z-Learner полягає у простому перетворенні цільової змінної, яке дозволяє перетворити довільну ймовірнісну класифікаційну модель у модель, що передбачає вплив. Варто зауважити, що на відміну від більшості методів моделей впливу, які використовують дві моделі, в цьому методі створюється єдина модель, яка безпосередньо моделює вплив, замість окремого моделювання ймовірностей тестової та контрольної груп.

Нехай:

1) $X \in \mathbb{R}^{n \times k}$ – значення k ознак, які описують n користувачів;

2) $W \in \mathbb{R}^{n \times 1}$ – ознака проведення комунікації, де $w_i = 1$ означає що проводилась комунікація з i-им користувачем, інакше 0;

3) $Y \in \mathbb{R}^{n \times 1}$ – ознака виконання цільової дії, де $y_i = 1$ означає що i-ий користувач виконав цільову дію, інакше 0.

Тоді для моделювання впливу змінна Y перетворюється в змінну Z за наступною формулою:

$$Z = \begin{cases} 1 & \text{якщо } (W = 1 \text{ та } Y = 1) \text{ або } (W = 0 \text{ та } Y = 0); \\ 0 & \text{інакше.} \end{cases}$$

Ідея перетворення полягає в тому, що Z дорівнює одиниці, якщо для певного випадку результат у тестовій групі був принаймні настільки ж хорошим, як у контрольній групі (якби для цього випадку було відомо результати у обох групах).

Умовну ймовірність події $Z = 1$ можна розписати наступним чином:

$$P(Z = 1|X) =$$

$$= P(Z = 1|X, W = 1)P(W = 1|X) + P(Z = 1|X, W = 0)P(W = 0|X) =$$

$$= P(Y = 1|X, W = 1)P(W = 1|X) + P(Y = 0|X, W = 0)P(W = 0|X)$$

Нехай для зручності формулювання $P^T(\dots|\dots) = P(\dots|\dots, W = 1)$ та $P^C(\dots|\dots) = P(\dots|\dots, W = 0)$. Будь-яке моделювання впливу будується на

припущенні, що змінні X та W незалежні, щоб модель під час тренування могла знайти справжні залежності без упереджености спричинених, наприклад, тренуванням моделі на даних, де в тестову групу користувачів відбирали за певним критерієм. Тоді $P(W|X) = P(W)$ і справедливо наступне:

$$P(Y = 1|X, W = 1)P(W = 1|X) + P(Y = 0|X, W = 0)P(W = 0|X) =$$
$$= P^T(Y = 1|X)P(W = 1) + P^C(Y = 0|X)P(W = 0)$$

Також для цієї моделі робиться додаткове припущення, що $P(W = 1) = P(W = 0) = \frac{1}{2}$, тобто користувачі розбиті на тестову та контрольну підгрупи порівну. Тоді:

$$P(Z = 1|X) = P^T(Y = 1|X)P(W = 1) + P^C(Y = 0|X)P(W = 0) =$$
$$= \frac{1}{2}\left(P^T(Y = 1|X) + P^C(Y = 0|X)\right) =$$
$$= \frac{1}{2}\left(P^T(Y = 1|X) + 1 - P^C(Y = 1|X)\right)$$

Таким чином:

$$P^T(Y = 1|X) - P^C(Y = 1|X) = 2P(Z = 1) - 1$$

Отже, для тренування моделі впливу достатньо перетворити змінну Y на Z та використовувати X як вхідні дані моделі, а Z як очікуваний результат. Для отримання оцінки впливу комунікації на користувачів необхідно використовувати наступну формулу: $uplift = 2P(Z = 1) - 1$ [1, 5].

2.3.3 Dependent Data Representation (DDR)

Метод моделювання впливу з залежним представленням даних (Dependent Data Representation, DDR) ґрунтується на методі ланцюгів класифікаторів [10], спочатку розробленому для проблем багатокласової класифікації.

Ідея полягає в тому, що якщо набір даних містить L різних класів, то можна побудувати L різних класифікаторів, кожен з яких вирішує проблему бінарної класифікації, і на етапі навчання кожен наступний класифікатор використовує передбачення попередніх як додаткові ознаки.

Оскільки методи моделювання впливу часто використовують дві моделі для оцінки впливу комунікації (одна передбачує ймовірність виконання цільової дії при наявності комунікації, інша за її відсутності), то було запропоновано використати аналогічну ланцюгам класифікаторів ідею в контексті моделювання впливу.

Нехай:

1) $X^C \in \mathbb{R}^{n\times k}, X^T \in \mathbb{R}^{m\times k}$ – значення k ознак, які описують n користувачів з контрольної (C – Control) та m тестової (T – Test) підгруп відповідно;

2) $Y^C \in \mathbb{R}^{n\times 1}, Y^T \in \mathbb{R}^{m\times 1}$ – ознака виконання цільової дії користувачами з контрольної та тестової підгруп відповідно, де $y_i = 1$ означає що i-ий користувач виконав цільову дію, 0 інакше.

Тоді алгоритм тренування DDR моделі впливу наступний.

1. Спочатку тренується перший ймовірнісний класифікатор $model^C$ на даних про контрольну групу:

$$model^C = fit\begin{pmatrix} x_{11} & \cdots & x_{1k} & y_1 \\ \vdots & \ddots & \vdots & \vdots \\ x_{n1} & \cdots & x_{nk}\text{'} & y_n \\ & X^C & & Y^C \end{pmatrix}$$

2. Дані про тестову групу та ймовірнісні передбачення натренованої $model^C$ на їх основі використовуються як вхідні дані при тренуванні $model^T$:

$$P^C = P\_model^C\begin{pmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mk} \\ & X^T & \end{pmatrix}$$

$$model^T = fit\begin{pmatrix} x_{11} & \cdots & x_{1k} & p_1^C & y_1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ x_{m1} & \cdots & x_{mk} & p_m^C, & y_m \\ & X^T & & P^C & Y^T \end{pmatrix}$$

Нехай $X \in \mathbb{R}^{l \times k}$ – значення k ознак, які описують l нових користувачів, тоді отримані вище моделі використовуються наступним чином для отримання оцінки впливу комунікації *uplift*:

$$uplift(X) = P\_model^T\big(X, P\_model^C(X)\big) - P\_model^C(X).$$

Даний метод можна аналогічно використати навпаки: спочатку натренувати ймовірнісний класифікатор на тесовій групі користувачів, а потім використовувати його ймовірнісні передбачення як ознаку для моделі, побудованої на контрольній групі користувачів [11].

2.4 Метрики оцінки якості моделі впливу

Фундаментальна складність моделювання впливу полягає в тому, що неможливо одночасно спостерігати реакцію користувача на наявність та відсутність взаємодії з ним. Тому неможливо точно визначити вид користувача: будь-який випадок може належати до двох класів. Наприклад, якщо відомо що користувач виконав цільову дію після комунікації з ним, то він може бути як схильним до переконання, так і лояльним користувачем. Так само вплив комунікації, який моделі впливу оцінюють, теж неможливо спостерігати. Таким чином, в моделюванні впливу не можна використовувати стандартні метрики оцінки якості моделей класифікації і для цього виду моделювання необхідно використовувати інші метрики, розроблені суто для нього [12].

В даній роботі використовуються наступні метрики: стовпчикова діаграма впливу за перцентильними рангами (uplift by percentile barchart), вплив на топ k% (uplift at top k%, для поточного аналізу розраховано на топ

30%), середньозважений вплив (weighted average uplift), крива Квіні (Qini curve) та коефіцієнт Квіні (Qini Coefficient).

2.4.1 Стовпчикова діаграма впливу за перцентильними рангами та середньозважений вплив

Стовпчикова діаграма впливу за перцентильними рангами будується за наступним алгоритмом.

1. Користувачі сортуються за спаданням спрогнозованого значення впливу.

2. Відсортовані дані діляться на перцентилі.

3. В кожному перцентилі окремо оцінюється вплив як різниця між середнім значенням цільової змінної в тестовій та контрольній групах [13].

Приклад стовпчикової діаграми впливу зображено на рисунку 2.3.

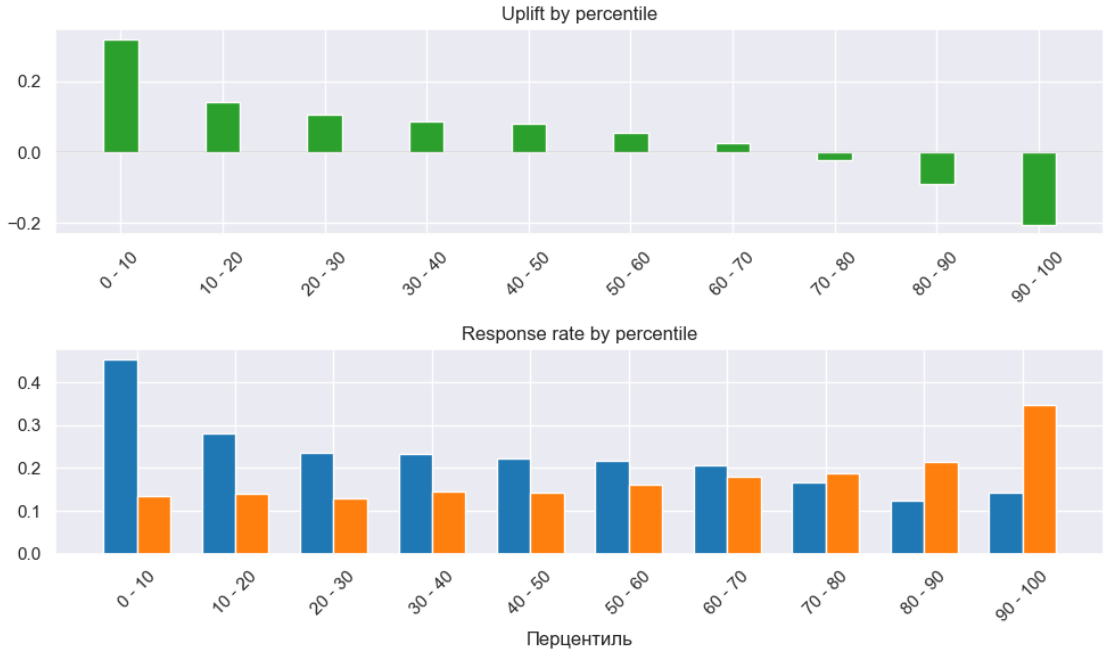


Рисунок 2.3. Приклад стовпчикової діаграми впливу за перцентильними рангами (синім значення для тестової групи, помаранчевим – контрольної)

Стовпчикова діаграма впливу допомагає зрозуміти чи правильно модель пріоритезує користувачів за впливом комунікації на них та в яких перцентилях є з цим проблеми. В ідеальному випадку ліва частина графіку відображає схильних до переконання користувачів, центральна лояльних та втрачених, а права сплячих собак, проте в залежності від акції певні види користувачів можуть бути відсутні на графіку [9].

Середньозважений вплив є числовим відображенням інформації зі стовпчикової діаграми впливу за перцентильними рангами та розраховується за наступною формулою:

$$Weighted\ average\ uplift = \frac{1}{\sum_{i=1}^{10} N_i^T} \sum_{i=1}^{10} N_i^T\ uplift_i, \text{ де}$$

- N_i^T – розмір тестової групи в i-ому перцентилі;

- $uplift_i$ – вплив в i-ому перцентилі.

Якщо ця метрика дорівнює 1, то в контрольній групі немає позитивних реакції у жодному перцентилі: користувачі ніколи не виконують цільову дію самостійно, а лише за умови комунікації. При такому значенні метрики немає сенсу розв'язувати задачу за допомогою моделювання підвищення, краще звести постановку задачі до навчання класичних моделі класифікації відгуку.

Значення метрики також може досягати граничного значення -1. Це відбувається, коли в тестовій групі немає реакцій $Y = 1$, а в контрольній групі всі клієнти мають реакцію $Y = 1$. Отже, значення метрики, які задовольняють для розв’язання задачі методами моделювання впливу лежать в межах (0, 1) [14].

2.4.2 Вплив на топ k%

Для розрахунку впливу на топ k% необхідно скористатись наступною формулою на топ k% користувачах за спрогнозованим значенням впливу:

$$Uplift\ at\ top\ k\% = \frac{Y^T_{top\ k\%}}{N^T_{top\ k\%}} - \frac{Y^C_{top\ k\%}}{N^C_{top\ k\%}}$$, де

- $Y^T_{top\ k\%}, Y^C_{top\ k\%}$ - кількість виконаних цільових дій в тестовій та контрольній групі відповідно серед топ k% користувачів за спрогнозованим значенням впливу;

- $N^T_{top\ k\%}, N^C_{top\ k\%}$ - кількість користувачів в тестовій та контрольній групі відповідно серед топ k% користувачів за спрогнозованим значенням впливу.

Ця метрика допомагає оцінювати вміння моделі пріоритезувати топ k% найкращих користувачів і корисна для випадків, коли бюджет рекламної кампанії розрахований на k% користувачів [15].

2.4.3 Крива та коефіцієнт Квіні

Для побудови кривої Квіні необхідно відсортувати дані за спаданням спрогнозованого значення впливу та побудувати графік за наступною формулою:

$$Qini\ curve(t) = Y^T_t - \frac{N^T_t}{N^C_t} Y^C_t$$, де

- t – кількість включених в комунікацію користувачів;

- Y^T_t, Y^C_t – кількість виконаних цільових дій в тестовій та контрольній групі відповідно;

- N^T_t, N^C_t – кількість користувачів в тестовій та контрольній групі відповідно.

Приклад кривої Квіні зображено на рисунку 2.4.

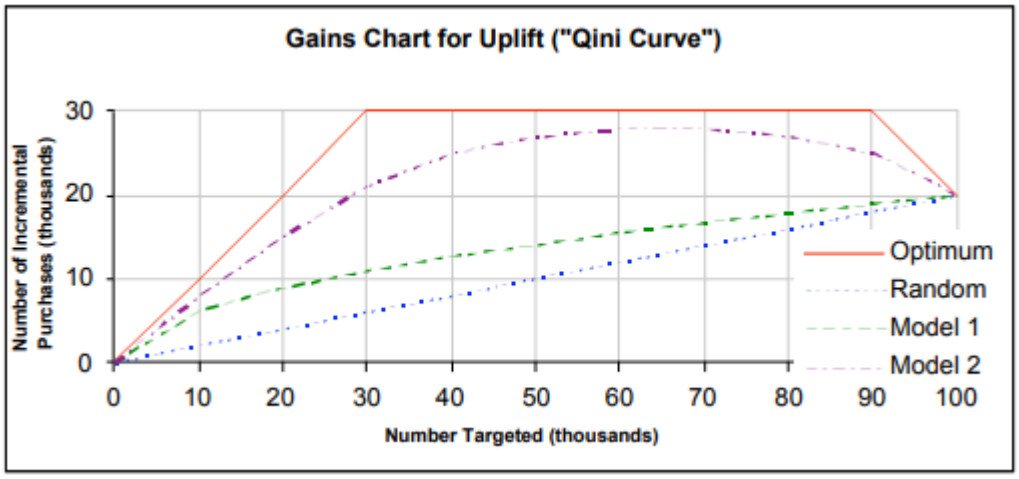


Рисунок 2.4. Приклад кривих Квіні. Червоним зображено криву оптимальної моделі для поточних даних, синім – випадкової моделі, зеленим та фіолетовим – побудованих моделей.

Для побудови кривої Квіні оптимальної моделі необхідно відсортувати дані за $uplift_{optimal} = Y * W - Y * (1 - W)$, де Y – ознака виконання цільової дії користувачами, а W – ознака наявності комунікації з користувачами. Таким чином оптимальною вважається модель, яка спочатку пріоритезує користувачів з Y=1 та W=1 (зробили цільову дію після комунікації), далі ідуть Y=0 (не зробили цільову дію) і в кінці Y=1 та W=0 (зробили цільову дію без комунікації).

Для побудови кривої Квіні випадкової моделі береться загальний вплив $uplift_{overall}$ рекламної кампанії на всій вибірці (відсоток позитивних відгуків) і будується пряма від точки $(0, 0)$ до $(n, uplift_{overall})$, де n – кількість учасників рекламної кампанії. Таким чином, крива Квіні випадкової моделі відображає середній вплив комунікації отриманий при випадковому розподілі користувачів на тестову та контрольні групи.

Формула сортування даних для оптимальної моделі пояснює частково суть кривої Квіні: вона нагороджує модель за вибір найкращими користувачів, які зробили цільову дію після комунікації, та штрафує за користувачів, які

зробили цільову дію без комунікації. Також множник $\frac{N_t^T}{N_t^C}$ в формулі кривої Квіні допомагає сильніше штрафувати модель за повернення високої оцінки впливу у випадках, де користувачі зробили цільову дію без комунікації, якщо кількість тестових користувачів більша за контрольних.

Коефіцієнт Квіні є числовим відображенням інформації з кривої Квіні розраховується за наступною формулою:

$$Qini\ coefficient = \frac{S_{model}}{S_{optimal}}, \text{ де}$$

- S_{model} – площа між кривими Квіні побудованої та випадкової моделей;
- $S_{optimal}$ – площа між кривими Квіні оптимальної та випадкової моделей.

Коефіцієнт Квіні дозволяє загалом оцінити потенціал моделі впливу в порівнянні з випадковою та оптимальною. Проте варто пам’ятати, що оптимальна модель є рідко досяжною в реальних випадках, тому криву та коефіцієнт Квіні варто використовувати як метрику для порівняння побудованих моделей, а не як метрику точності моделей [16, 17].

2.5 Висновки до розділу 2

В даному розділі була визначена постановка завдання моделювання впливу у контексті рекламної кампанії, що визначило основні цілі та напрями дослідження. Також розглянуто види користувачів у моделюванні впливу, що дозволило розширити уявлення про їхні особливості та підкреслило переваги моделювання впливу в порівнянні зі звичайними класифікаційними моделями відгуку.

Зроблено огляд різних моделей впливу, таких як S-Learner, Z-Learner та Dependent Data Representation (DDR), що надають різноманітні підходи до

моделювання впливу у рекламних кампаніях, визначено їх особливості та необхідні припущення.

Останній підрозділ стосувався розгляду різних метрик оцінки ефективності моделей впливу, таких як стовпчикова діаграма впливу за перцентильними рангами, середньозважений вплив, вплив на топ k%, крива та коефіцієнт Квіні, які будуть важливими та корисними інструментами для порівняльного аналізу моделей та в реалізованій СППР.

РОЗДІЛ 3 ПОРІВНЯННЯ МЕТОДІВ МОДЕЛЮВАННЯ ВПЛИВУ ТА РОЗРОБКА СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕННЯ

Метою даного розділу є порівняння DDR методу моделювання впливу з іншими моделями, розглянутими в розділі 2, та розробка СППР на їх основі.

3.1 Порівняльний аналіз обраного методу моделювання з іншими

Метод залежного представлення даних визначається принципом надання моделям додаткової інформації з метою покращення їхньої точності та ефективності. Однак перед розробкою системи підтримки прийняття рішень, необхідно з'ясувати, чи цей метод є оптимальним та корисним для конкретної задачі. Для досягнення цієї мети, важливо провести порівняльний аналіз методу залежного представлення даних з іншими підходами моделювання впливу. Цей аналіз допоможе визначити переваги та обмеження методу перед його впровадженням у систему прийняття рішень.

3.1.1 Методика порівняльного аналізу

Для порівняння обраного методу з іншими в першу чергу треба визначити методику порівняння, що дозволить об'єктивно оцінити кожну модель та визначити їх слабкі та сильні сторони. Методика включає наступні кроки.

1. Визначення метрик ефективності.
2. Вибір методів для порівняння.

3. Збір та підготовка даних.

4. Навчання, тестування та порівняння моделей.

5. Висновки.

В рамках аналізу зосереджено увагу на метриках оцінки ефективності моделі таких як стовпчикова діаграма впливу за перцентильними рангами, вплив на топ 30%, середньозважений вплив, крива Квіні та коефіцієнт Квіні.

Для порівняння використано методи моделювання впливу S-Learner та Z-Learner. Оскільки всі методи, що порівнюються, є методами мета-навчання, які використовують звичайні ймовірнісні моделі класифікації з певним принципом перетворення вхідних або вихідних даних для створення моделей впливу, то для справедливості порівняння всі методи будуть порівнюватись на одному виді моделей класифікації - XGBoost класифікаторі.

Для тренування та порівняння моделей взято анонімізований набір даних телекомунікаційної компанії [18], який містить в собі наступні ознаки:

- id – ідентифікатор користувача;

- X_1, …, X_50 – 50 анонімізованих ознак, які описують користувача;

- treatment_group – до якої групи відноситься користувач (тестова чи контрольна);

- conversion – чи зробив користувач цільову дію.

Дані розбито випадковим чином на тренувальну (розміром 480 тис. рядків) та тестову (розміром 60 тис. рядків) вибірки. Розподіл на контрольну та тестову групи складає 50% та 20% користувачів виконали цільову дію. При цьому в тестовій групі 22.9% користувачів виконали цільову дію, а в контрольній 17.94%.

Для аналізу даних використано коефіцієнт рангової кореляції Спірмена, який на відміну від коефіцієнта кореляції Пірсона шукає монотонну залежність між даними, а не тільки лінійну [19]. Кореляційна матриця тренувальних даних зображена на рисунку 3.1.

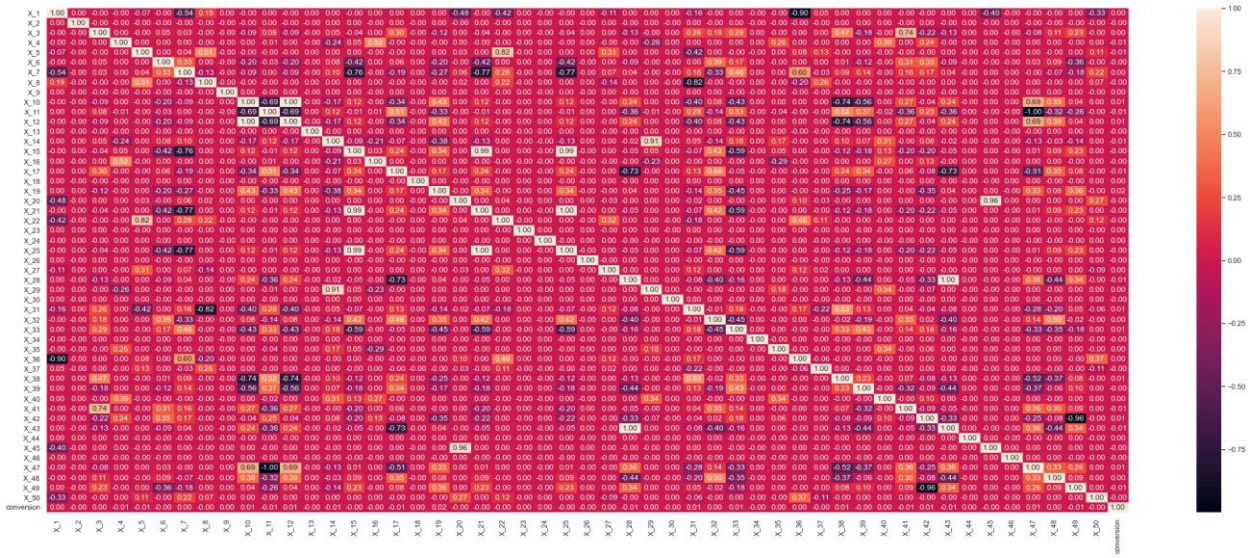


Рисунок 3.1. Кореляційна матриця Спірмена на основі тренувальних даних

В тренувальних даних було помічено 16 пар змінних, які мали високу кореляції між собою (більше 0.7). Для кожної пари було визначено змінну, що має меншу за модулем кореляцію з цільовою змінною, та видалено її. Після очистки даних отримали кореляційну матрицю зображену на рисунку 3.2.

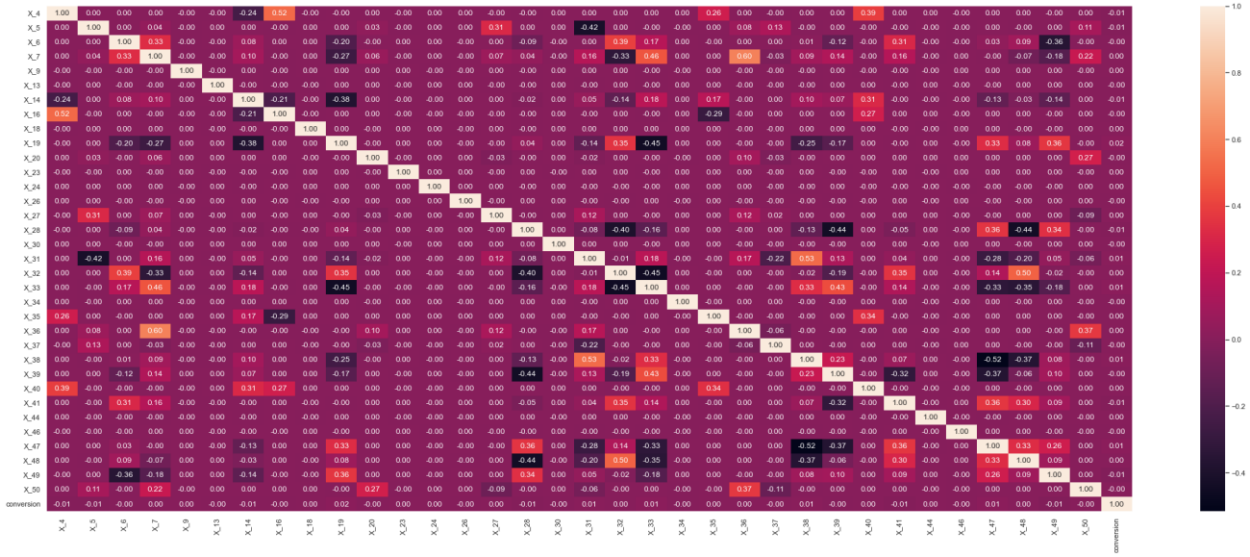


Рисунок 3.2. Кореляційна матриця Спірмена на основі тренувальних даних після прибирання змінних з високим коефіцієнтом кореляції

Після внесення аналогічних змін у тестові дані можна переходити до тренування та порівняння методів.

3.1.2 Результати порівняльного аналізу

Після тренування моделей було розраховано вищезгадані метрики оцінки якості кожної моделі на тестових даних. Результати методу S-Learner знаходяться в таблиці 3.1 та рисунках 3.3 і 3.4.

Таблиця 3.1 – Табличні результати методу S-Learner

Назва Методу	Вплив на топ 30%	Середньозважений вплив	Коефіцієнт Квіні
S-Learner	0.18825	0.04996	0.21522

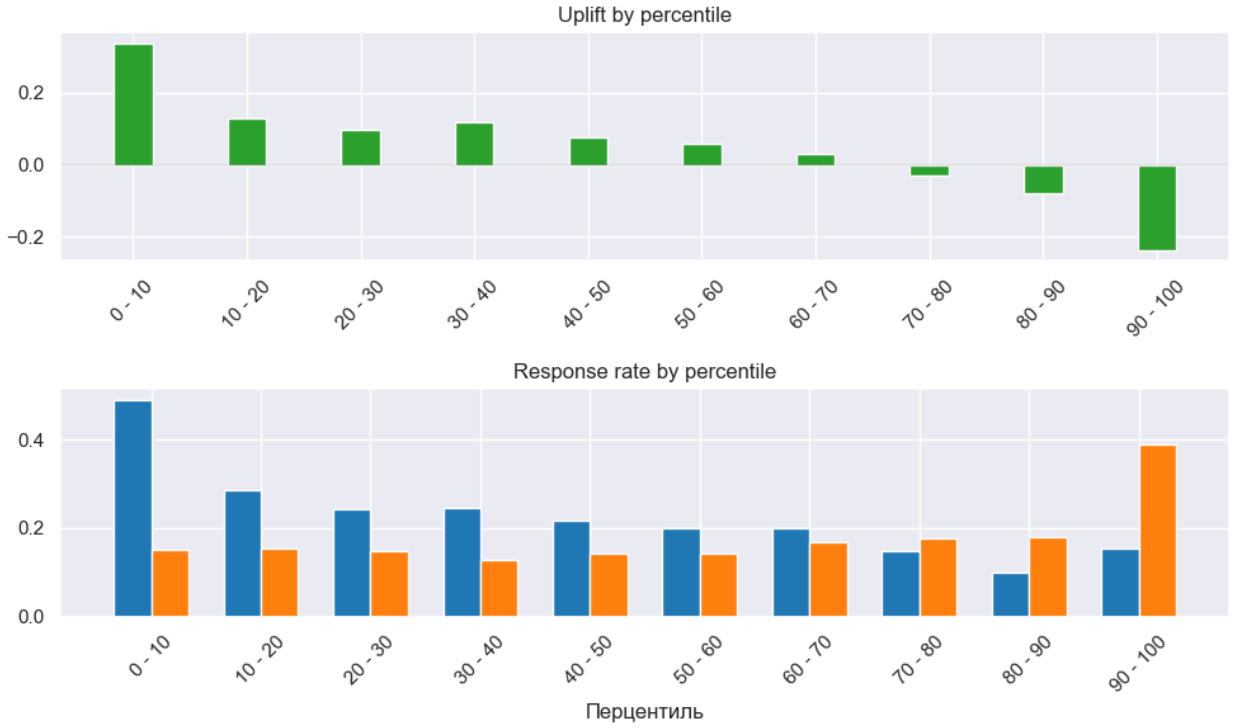


Рисунок 3.3. Стовпчикова діаграма впливу за перцентильними рангами методу S-Learner

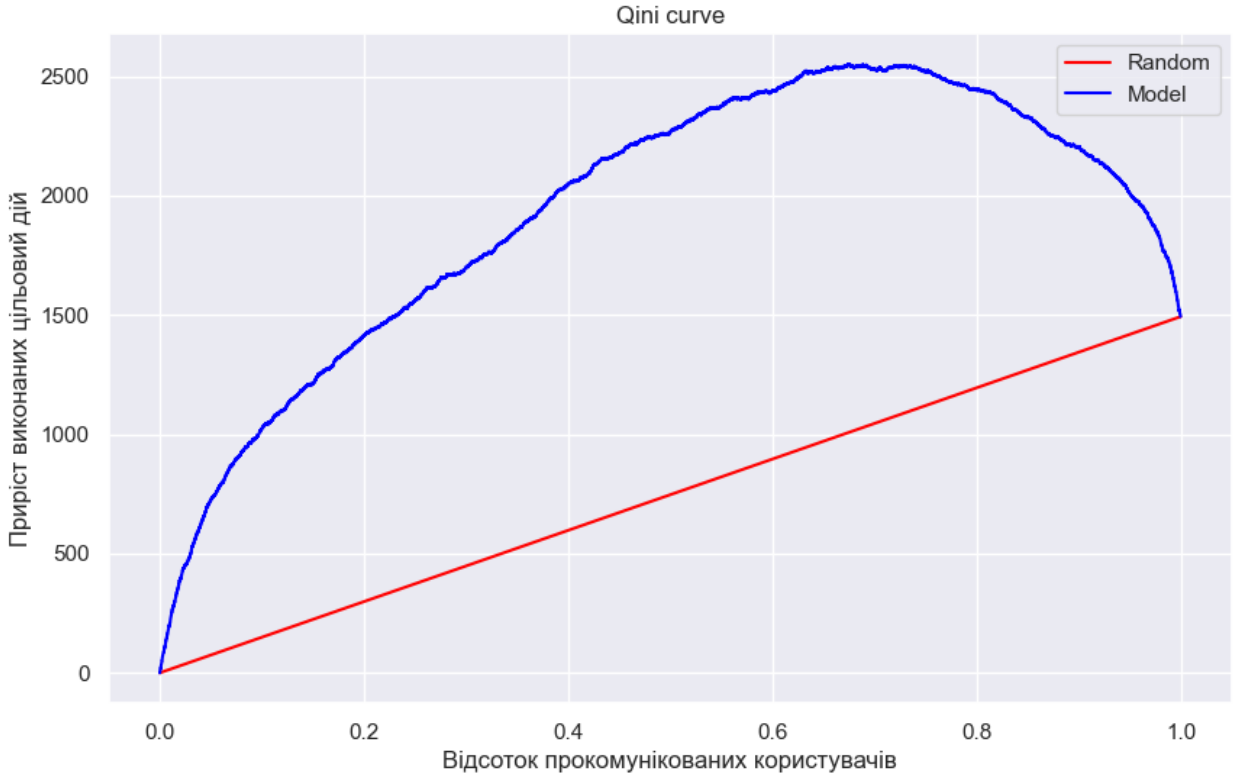


Рисунок 3.4. Графік кривої Квіні методу S-Learner

Загалом отримано непогані результати для методу S-Learner, проте при аналізі стовпчикової діаграми впливу за перцентильними рангами можна помітити, що значення впливу 30-го перцентиля нижче 40-го. Це значить що за результатами моделі в 30-ий перцентиль або потрапило менше користувачів з тестової групи, що виконали цільову дію, або більше користувачів з контрольної групи, які виконали цільову дію, ніж в 40-ий перцентиль. Тобто модель має проблеми з пріоритезацією користувачів за впливом в рамках 30-го та 40-го перцентилів [20].

Результати методу Z-Learner знаходяться в таблиці 3.2 та рисунках 3.5 і 3.6.

Таблиця 3.2 – Табличні результати методів S-Learner та Z-Learner

Назва Методу	Вплив на топ 30%	Середньозважений вплив	Коефіцієнт Квіні
S-Learner	0.18825	0.04996	0.21522
Z-Learner	0.18951	0.04989	0.20253

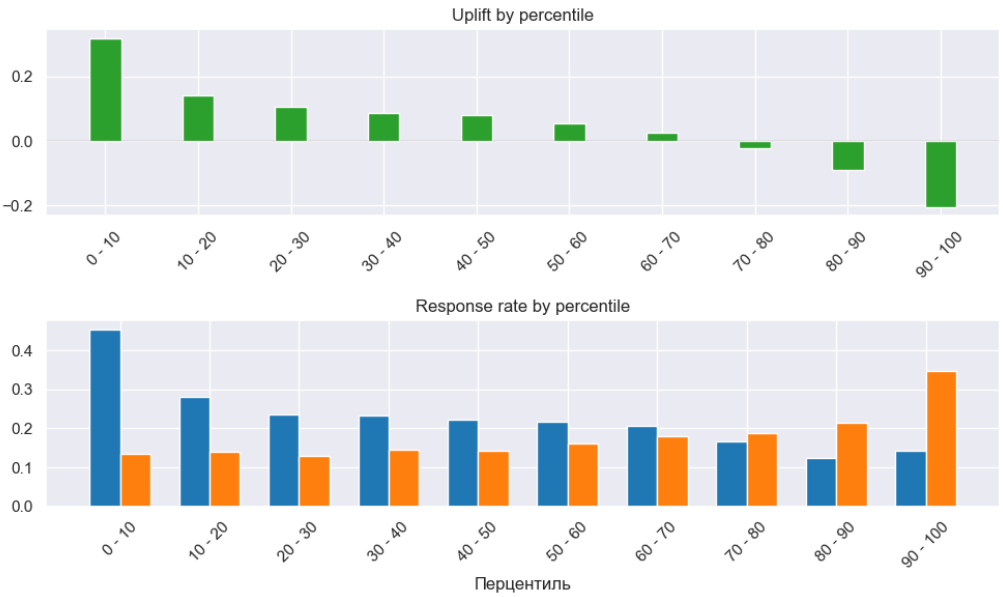


Рисунок 3.5. Стовпчикова діаграма впливу за перцентильними рангами методу Z-Learner

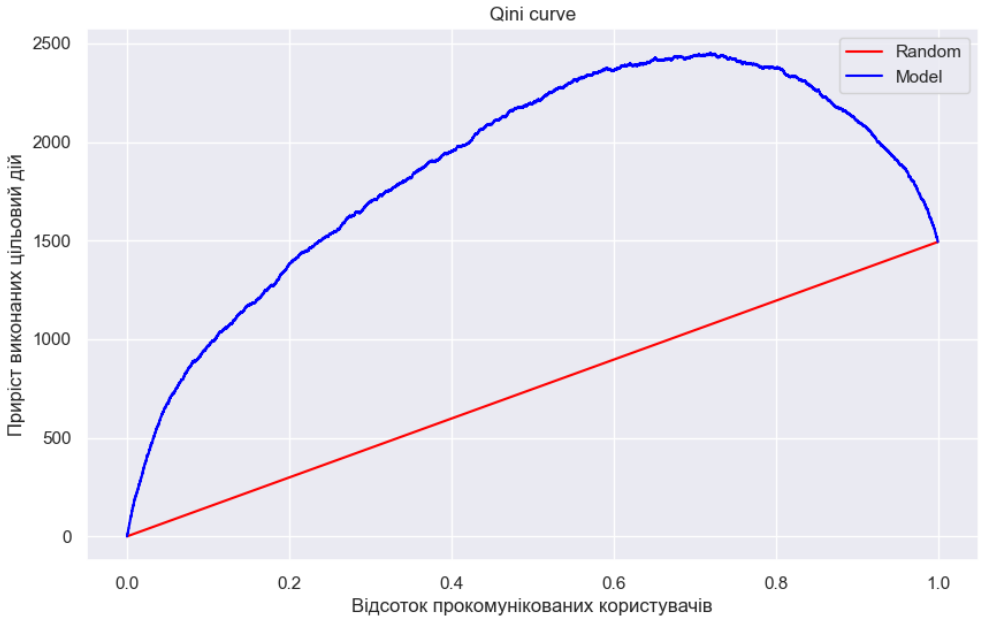


Рисунок 3.6. Графік кривої Квіні методу Z-Learner

Z-Learner трохи краще визначає топ 30% найкращих користувачів для комунікації і немає проблеми з перцентилями, які були у S-Learner. Проте гірші значення середньозваженого впливу та коефіцієнту Квіні свідчать про те, що на всьому обсязі тестових даних Z-Learner гірше справляється з оцінкою впливу комунікації на користувачів.

Результати методу DDR з використанням результатів моделі контрольної групи в моделі тестової групи знаходяться в таблиці 3.3 та рисунках 3.7 і 3.8.

Таблиця 3.3 – Табличні результати методів S-Learner, Z-Learner та DDR з використанням результатів моделі контрольної групи в моделі тестової групи

Назва Методу	Вплив на топ 30%	Середньозважений вплив	Коефіцієнт Квіні
S-Learner	0.18825	0.04996	0.21522
Z-Learner	0.18951	0.04989	0.20253
DDR(feature='control')	0.20224	0.04926	0.22559

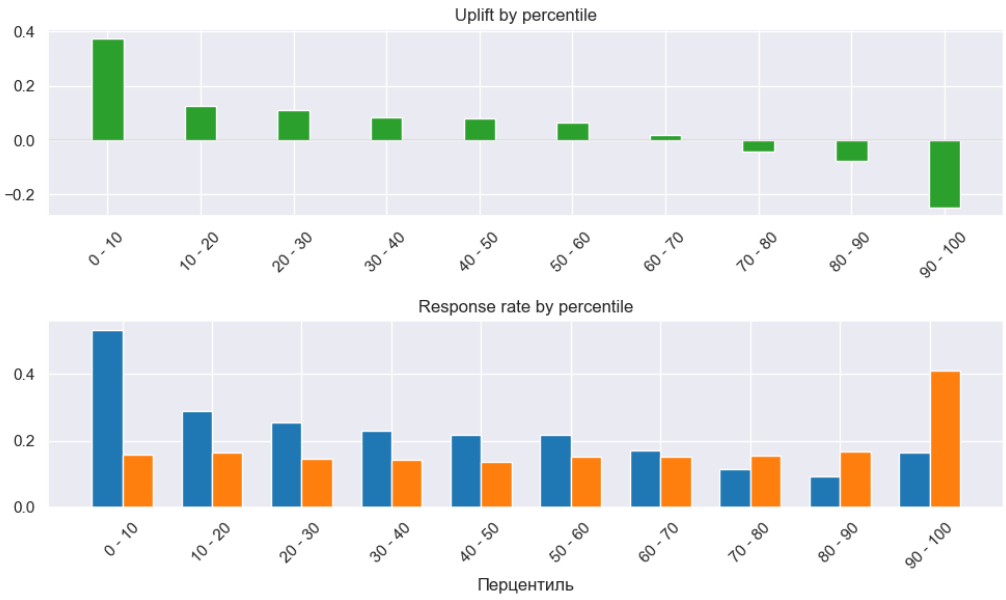


Рисунок 3.7. Стовпчикова діаграма впливу за перцентильними рангами методу DDR з використанням результатів моделі контрольної групи в моделі тестової групи

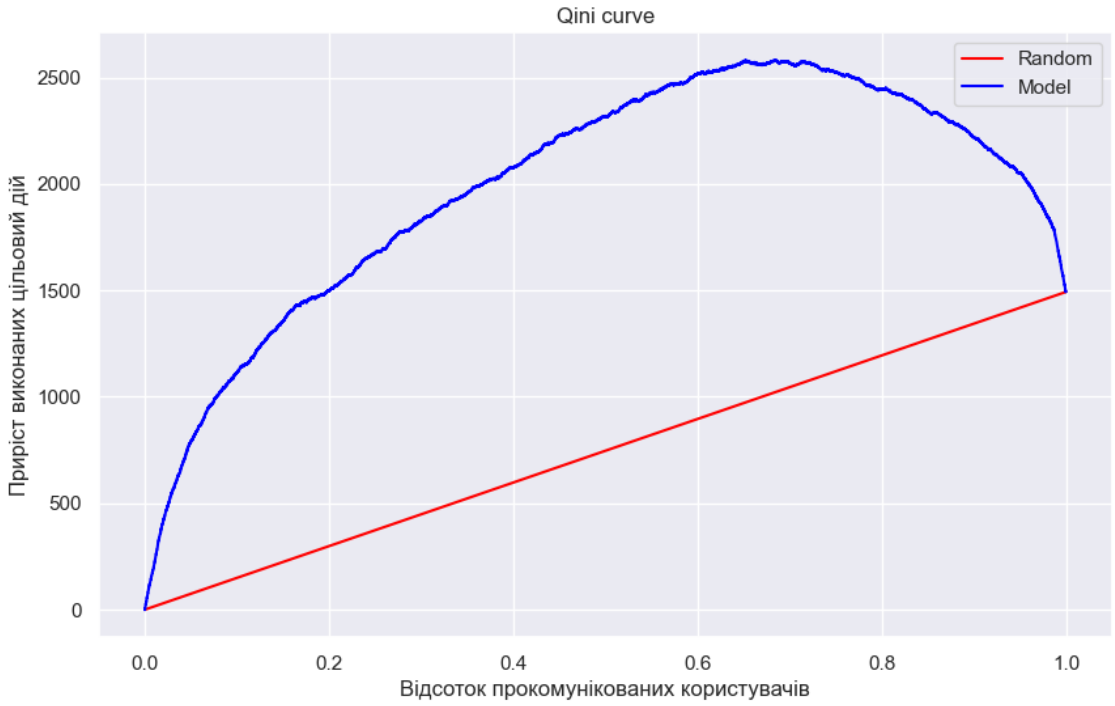


Рисунок 3.8. Графік кривої Квіні методу DDR з використанням результатів моделі контрольної групи в моделі тестової групи

Метод DDR з використанням результатів моделі контрольної групи в моделі тестової групи має найгірше значення середньозваженого впливу та найкращі значення коефіцієнту Квіні та впливу на топ 30% користувачів серед розглянутих методів. Також на стовпчиковій діаграмі можна помітити що впливи 10-го та 30-го перцентилів збільшилися, а 20-го трохи зменшився в порівнянні з минулим методами. Можна прийти до висновку, що загалом модель показала кращі результати, особливо на топ 30% користувачів.

Результати методу DDR з використанням результатів моделі тестової групи в моделі контрольної групи знаходяться в таблиці 3.4 та рисунках 3.9 і 3.10.

Таблиця 3.4 – Табличні результати методів S-Learner, Z-Learner та обох варіантів DDR (червоним – найгірші значення, зеленим – найкращі)

Назва Методу	Вплив на топ 30%	Середньозважений вплив	Коефіцієнт Квіні
S-Learner	0.18825	0.04996	0.21522
Z-Learner	0.18951	0.04989	0.20253
DDR(feature='control')	0.20224	0.04926	0.22559
DDR(feature='treatment')	0.20367	0.05023	0.22279

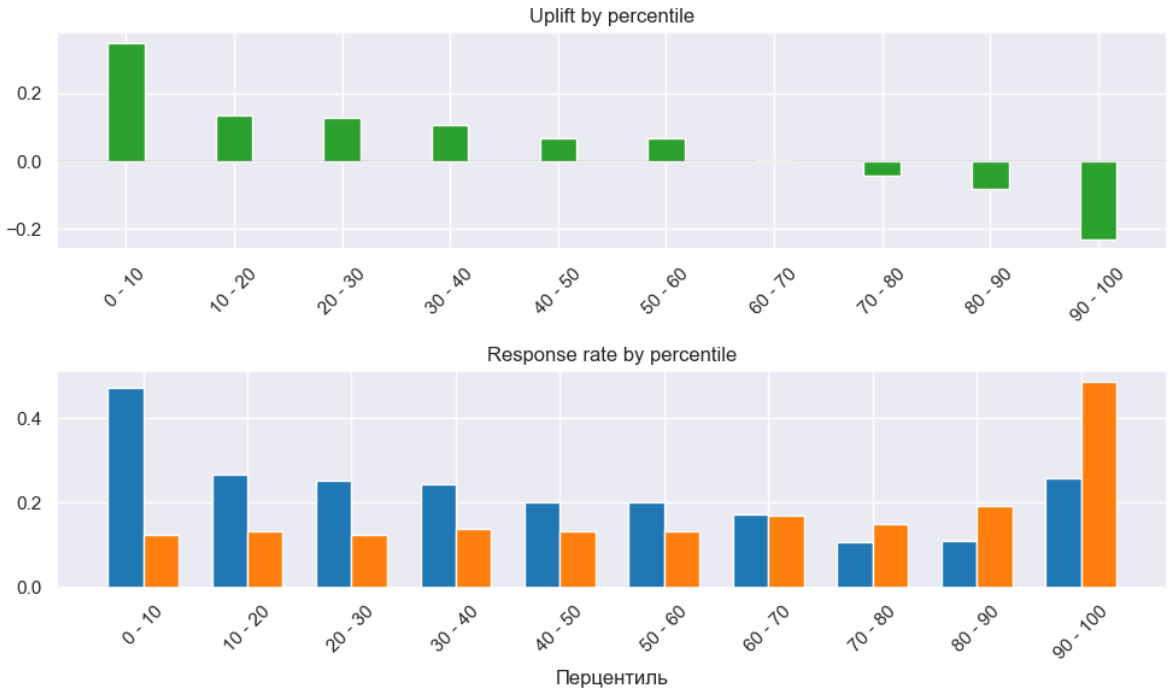


Рисунок 3.9. Стовпчикова діаграма впливу за перцентильними рангами методу DDR з використанням результатів моделі тестової групи в моделі контрольної групи

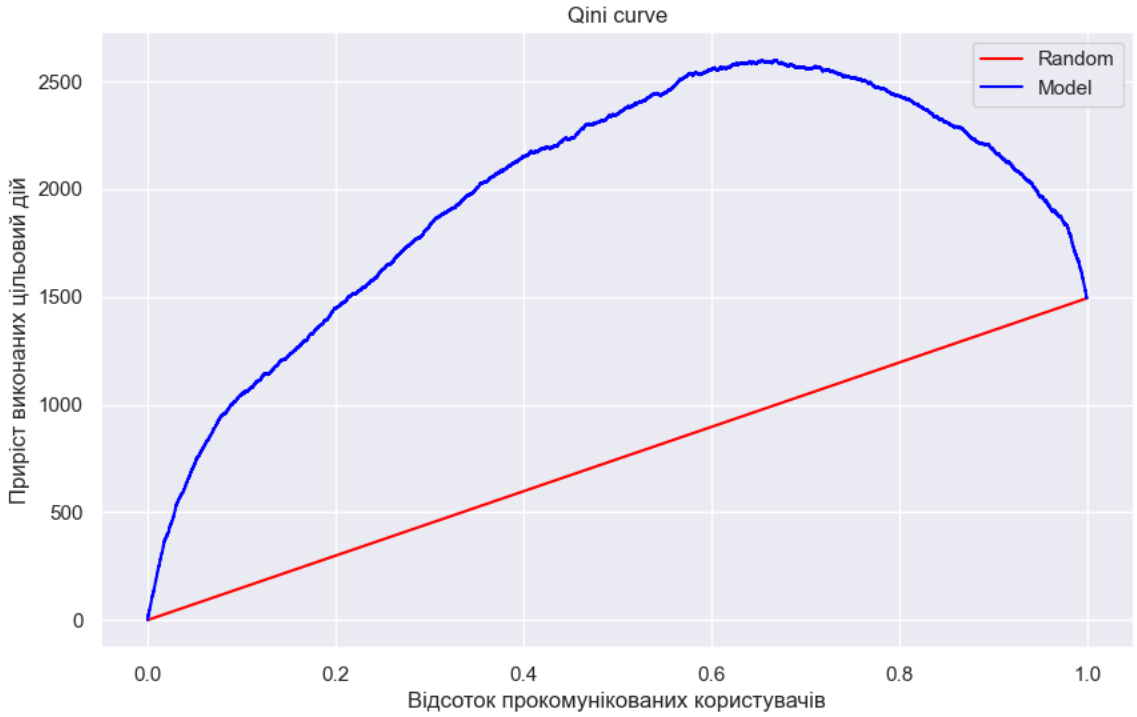


Рисунок 3.10. Графік кривої Квіні методу DDR з використанням результатів моделі тестової групи в моделі контрольної групи

Метод DDR з використанням результатів моделі тестової групи в моделі контрольної групи отримав найкращі значення впливу на топ 30% користувачів та середньозваженого впливу, а також друге найвище значення коефіцієнту Квіні. Це робить його найкращим методом серед усіх розглянутих для поточного набору даних.

3.1.3 Висновки порівняльного аналізу

Проведено порівняльний аналіз результатів обраного методу моделювання впливу з іншими на обраному наборі даних. З отриманими результатами можна ознайомитись в таблиці 3.4 та на рисунках 3.11 і 3.12.

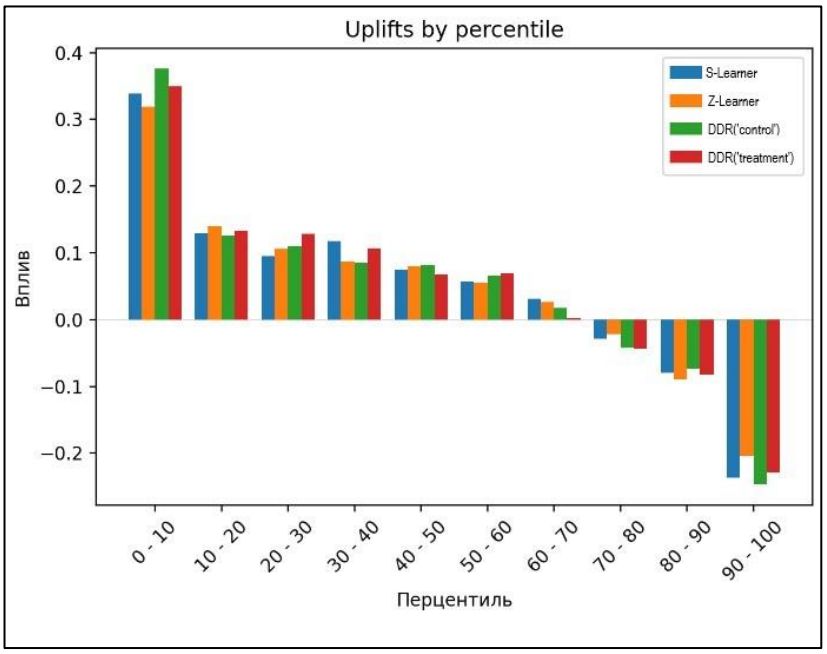


Рисунок 3.11. Графік впливів побудованих моделей за перцентильними рангами

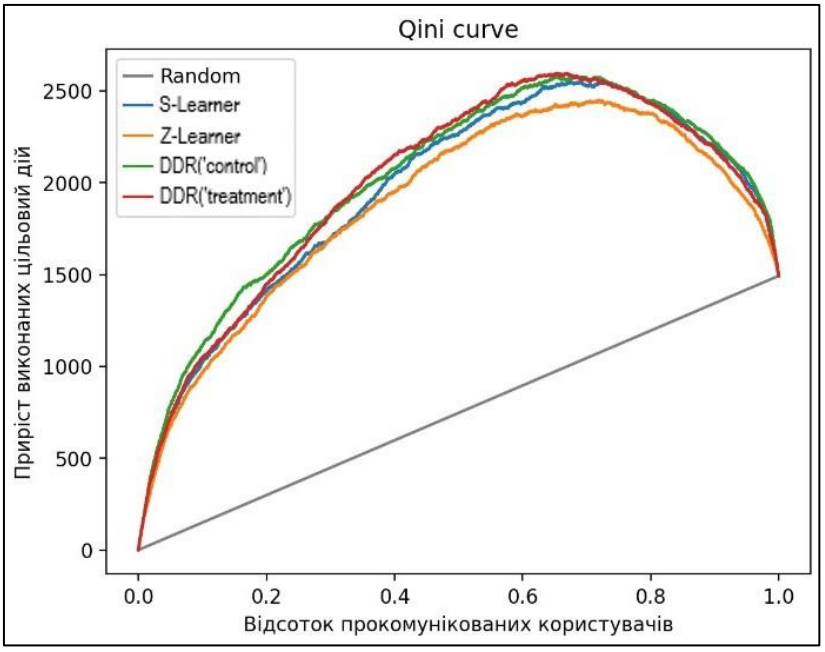


Рисунок 3.12. Графік кривої Квіні для побудованих моделей

Порівняльний аналіз показав, що обидва варіанти обраного методу DDR показали кращі результати за інші розглянуті методи. Опираючись на результати, можна прийти до висновку, що якщо бюджет рекламної кампанії

розрахований до 30% користувачів, то для розглянутого набору даних краще використовувати метод DDR з використанням результатів моделі контрольної групи в моделі тестової групи. У всіх інших випадках, краще використовувати метод DDR з використанням результатів моделі тестової групи в моделі контрольної групи для розглянутого набору даних.

3.2 Розробка системи підтримки прийняття рішень на основі методу моделювання впливу з залежним представленням даних

3.2.1 Вимоги та середовище розробки СППР

При створенні до СППР було висунуто наступні умови.

1. Інтерактивність - система повинна мати інтуїтивно зрозумілий графічний інтерфейс користувача (GUI), який дозволяє користувачам взаємодіяти з системою за допомогою кнопок, меню, графічних елементів тощо.

2. Доступність – система повинна бути легкодоступною для користувачів. Чим менше користувачу потрібно зробити для початку користування продуктом, тим більше ймовірності привернути його увагу.

3. Повнота – в системі має бути можливість провести повноцінний аналіз впливу: починаючи від завантаження даних для тренування моделі, закінчуючи використанням нових даних на створеній моделі та вивантаженням отриманих результатів.

Враховуючи вищезгадані вимоги, було прийнято рішення розробляти програмний продукт на мові програмування Python. Основними перевагами цієї мови програмування є наступні.

1. Простота та читабельність коду: Python відомий своєю простотою та читабельністю. Це робить його ідеальним вибором для розробки програм,

оскільки код на Python легко розуміти і підтримувати. Це особливо важливо в машинному навчанні, де зрозумілий код сприяє розробці та налагодженню моделей.

2. Багата екосистема бібліотек: Python має велику кількість бібліотек та фреймворків для машинного навчання, таких як TensorFlow, Keras, PyTorch, scikit-learn, NumPy, Pandas, і багато інших. Це дозволяє розробникам швидко створювати та навчати моделі.

3. Загальне призначення: Python є мовою загального призначення, тобто його можна використовувати для різних завдань. Він підходить для розробки веб-додатків, наукових досліджень, обробки даних, машинного навчання та інших завдань.

Для створення інтерфейсу використано функціонал бібліотеки Streamlit. Streamlit - це потужний інструмент для розробки інтерфейсів програм, який має наступні переваги.

1. Простота використання: Streamlit розроблений з урахуванням простоти та легкості використання. Він дозволяє створювати інтерфейси програм, використовуючи Python, без необхідності вивчення складних мов програмування або інших інструментів. Це особливо корисно для даних науковців та інших, хто має обмежений досвід веб-розробки, тому що дозволяє їм більше сфокусуватись на дослідженні замість реалізації інтерфейсу.

2. Інтерактивність: Streamlit надає можливість створювати інтерактивні інтерфейси, де користувачі можуть взаємодіяти з програмою, міняти параметри та бачити результати в реальному часі. Це ідеально підходить для демонстрації аналітики даних, моделей машинного навчання та інших складних операцій.

3. Швидкість розробки: Streamlit дозволяє розробникам створювати прототипи та паралельно тестувати їх в реальному часі, що особливо корисно для експериментів та тестування концепцій.

4. Створення веб-застосунків: Streamlit дозволяє з легкістю вивантажувати розроблену програму у вільний доступ у вигляді веб-застосунку. Це покриває вимогу доступності СППР для користувачів.

5. Здатність до візуалізації даних: Streamlit підтримує вбудовані інструменти для створення графіків, таблиць, та інших елементів візуалізації. Це робить його відмінним вибором для розробки веб-інтерфейсів для аналітики даних та візуалізації результатів.

6. Безкоштовний та відкритий код: Streamlit є безкоштовним та відкритим програмним забезпеченням та має активну спільноту розробників, які надають підтримку та розробляють додатки.

Блок-схема алгоритму роботи з СППР зображена на рисунку 3.13.

Рис. 3.13. Блок-схема алгоритму роботи з СППР

UML-діаграма класів СППР зображена на рисунку 3.14.

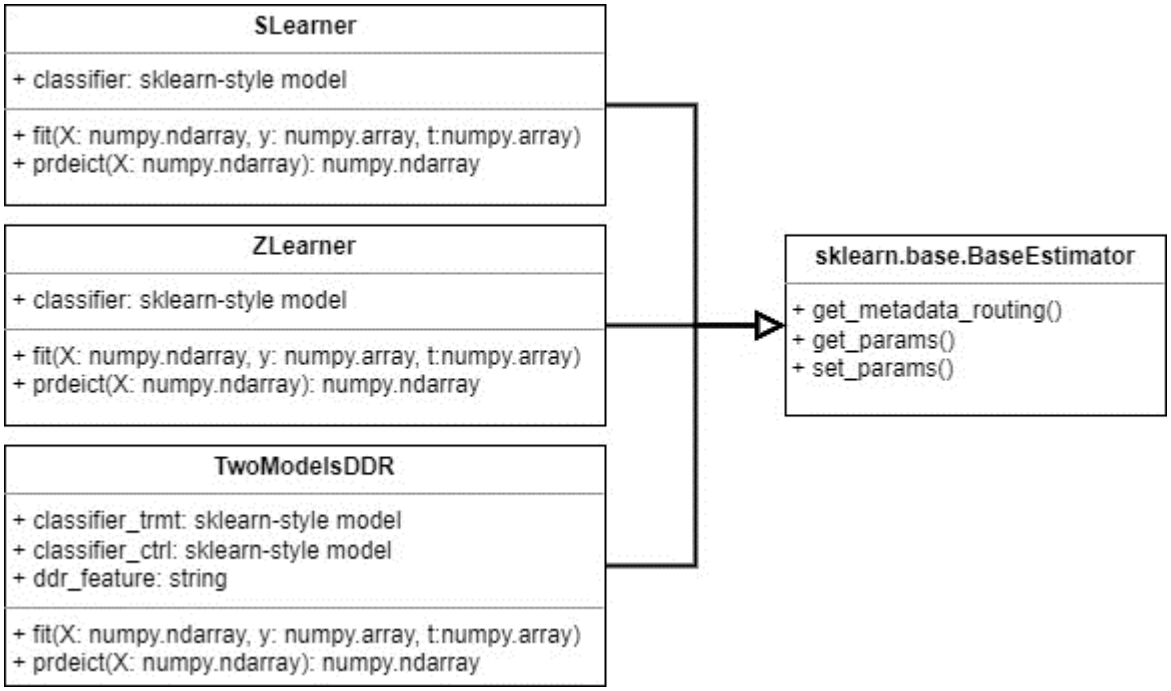


Рис. 3.14. UML-діаграма класів СППР

Опис класів, зображених на рисунку 3.14:

- sklearn.base.BaseEstimator – клас, що містить базові методи для створення власних класів моделей;

- SLearner – клас власноруч реалізованого методу моделювання впливу S-Learner;

- ZLearner – клас власноруч реалізованого методу моделювання впливу Z-Learner;

- TwoModelsDDR – клас власноруч реалізованого методу моделювання впливу DDR.

3.2.2 Інтерфейс програмного продукту

Для ознайомлення з інтерфейсом розробленої СППР пройдемо повний шлях моделювання впливу від завантаження даних до вивантаження результатів моделі на нових даних для комунікації.

На рисунку 3.15 зображено початкове вікно програми.

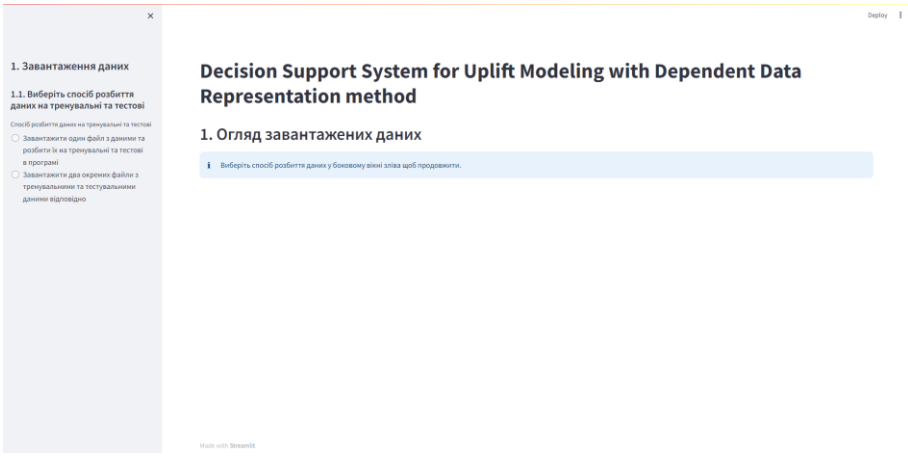


Рисунок 3.15. Початкове вікно програми

Після вибору завантаження двох наборів даних, програма надає кнопки для пошуку необхідних файлів на пристрої та підказує які формати файлів дозволені (рис. 3.16).

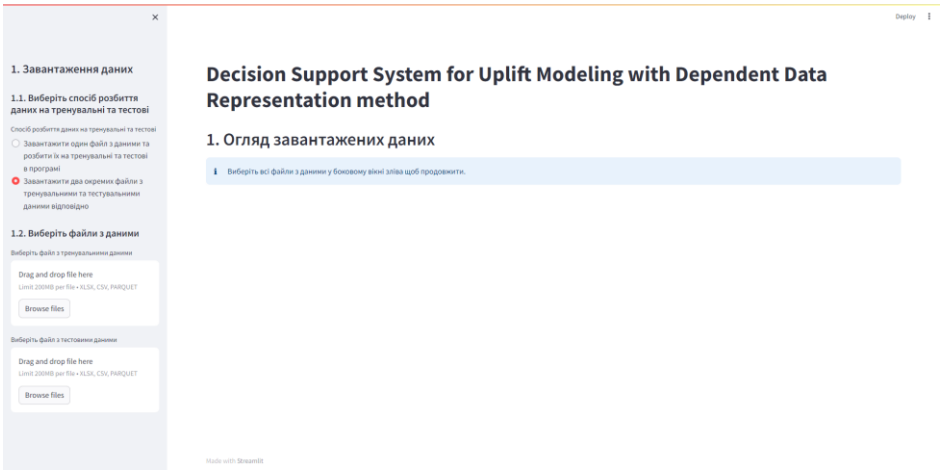


Рисунок 3.16. Можливість вибору вхідних файлів

Після завантаження даних, програма надає можливість ознайомитись з ними (рис. 3.17 та 3.18). Далі необхідно визначити для СППР ознаки, які відображають ідентифікатор користувача, ознаку комунікації та ознаку виконання цільової дії (рис. 3.18).

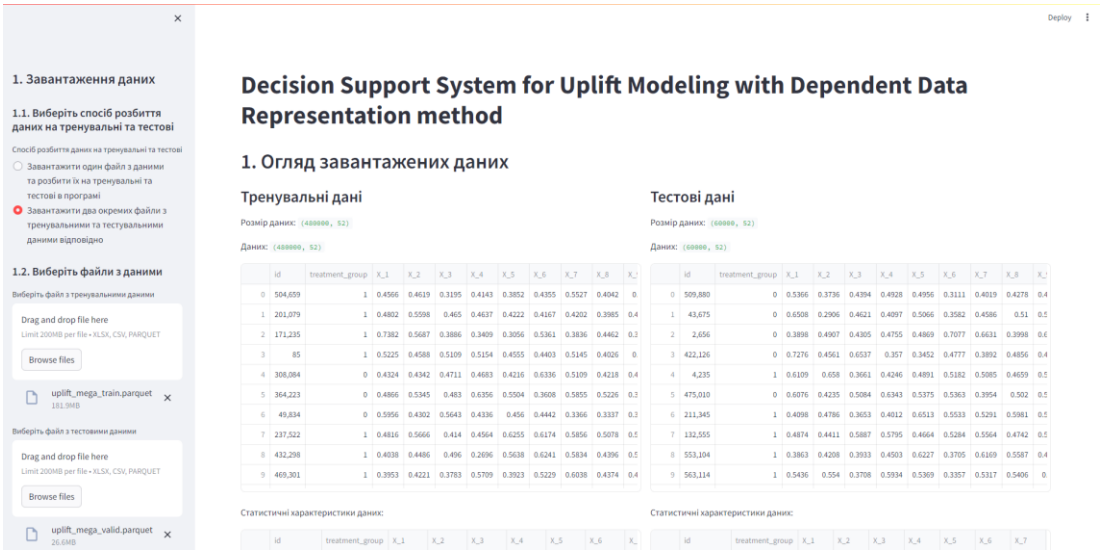


Рисунок 3.17. Ознайомлення з завантаженими даними

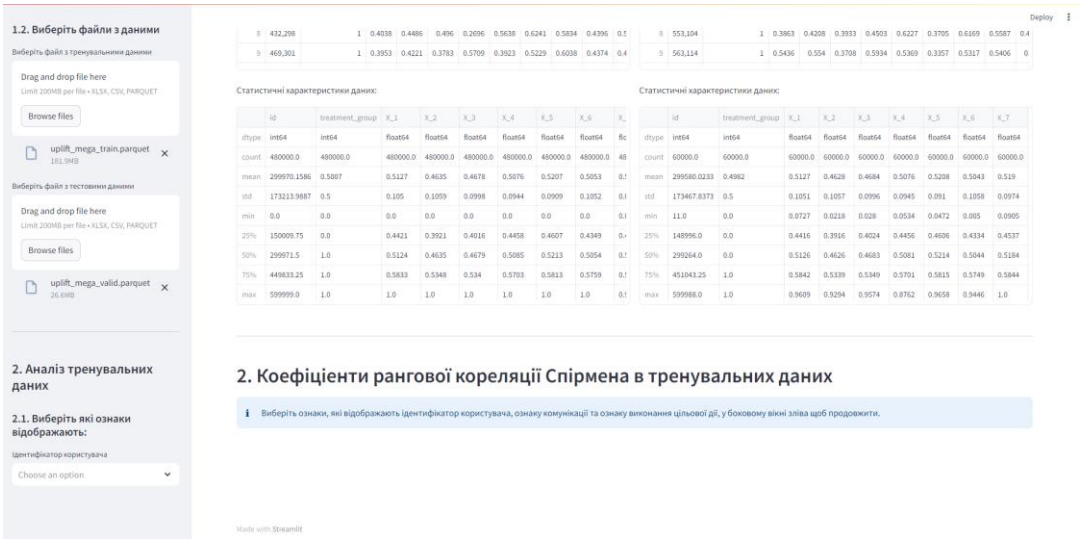


Рисунок 3.18. Продовження ознайомлення з завантаженими даними

Після визначення вищезгаданих ознак, СППР пропонує згенерувати кореляційну матрицю Спірмена (рис. 3.19).

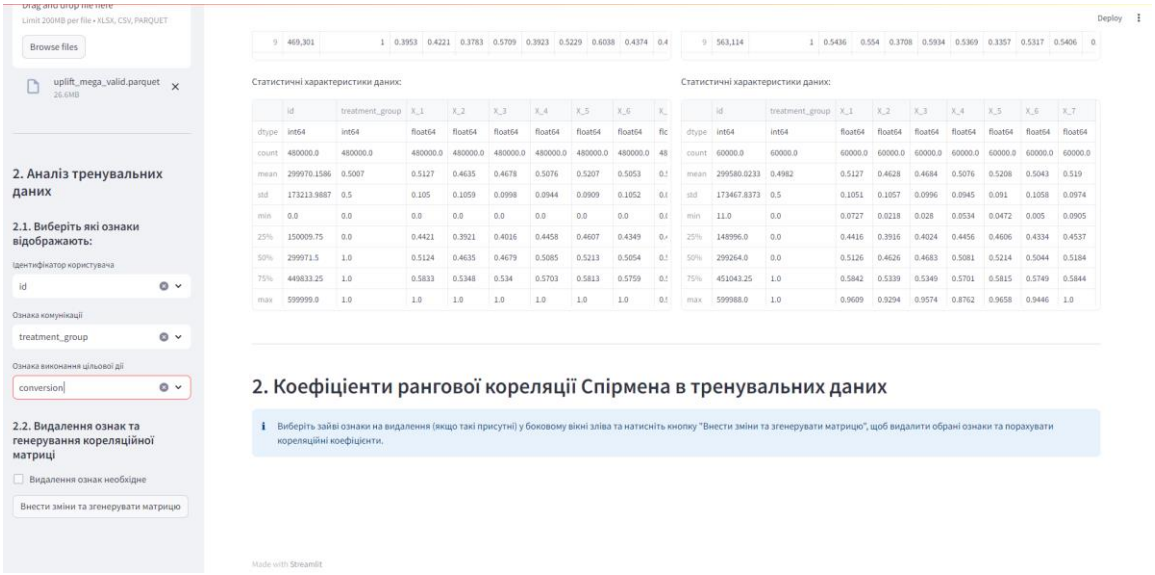


Рисунок 3.19. Кнопка для генерування кореляційної матриці

Після натискання кнопки генерується кореляційна матриця, яку можна розширити на весь екран та наведення на елементи показує інформацію про обраний перетин змінних. Також виводяться коефіцієнти рангової кореляції Спірмена всіх ознак з цільовою змінною для усієї тренувальної вибірки (рядок conversion), для тестової групи (рядок conversion_treatment) та для контрольної групи (рядок conversion_control, рис. 3.20).

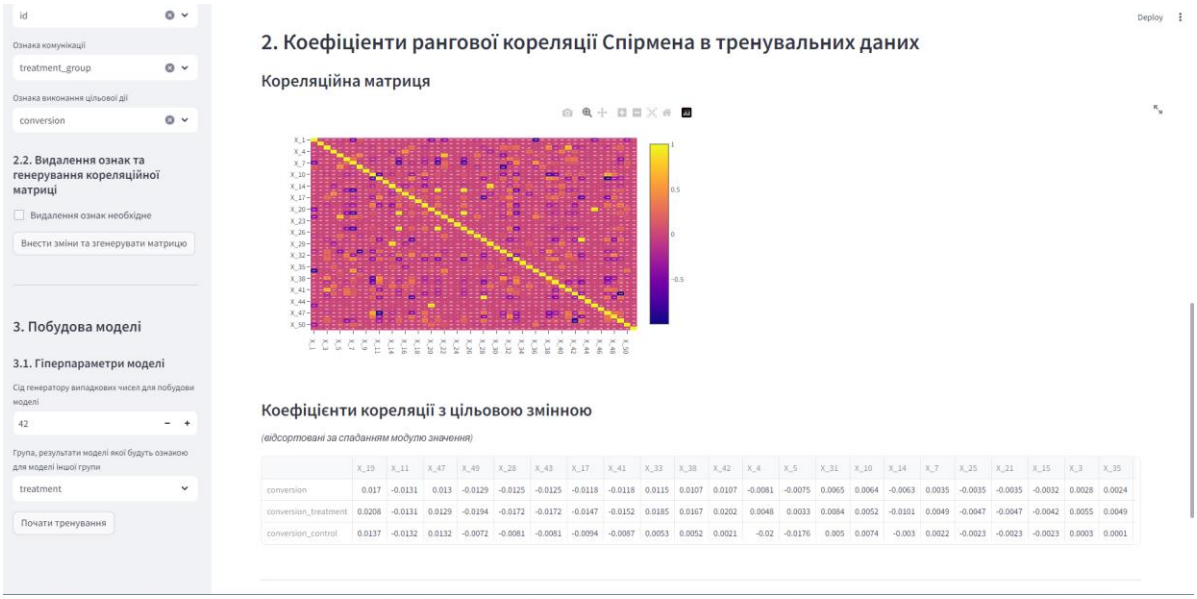


Рисунок 3.20. Кореляційна матриця

Оскільки дані містять пари змінних з високою кореляцією, то видаляємо їх аналогічно процесу, описаному в порівняльному аналізі. Після вибору змінних на видалення натискаємо повторно кнопку «Внести зміни та згенерувати матрицю», щоб видалити обрані ознаки та переконатись у внесенні змін (рис. 3.21).

Рисунок 3.21. Кореляційна матриця після видалення ознак

Нижче СППР пропонує обрати методу моделювання впливу (рис. 3.22).

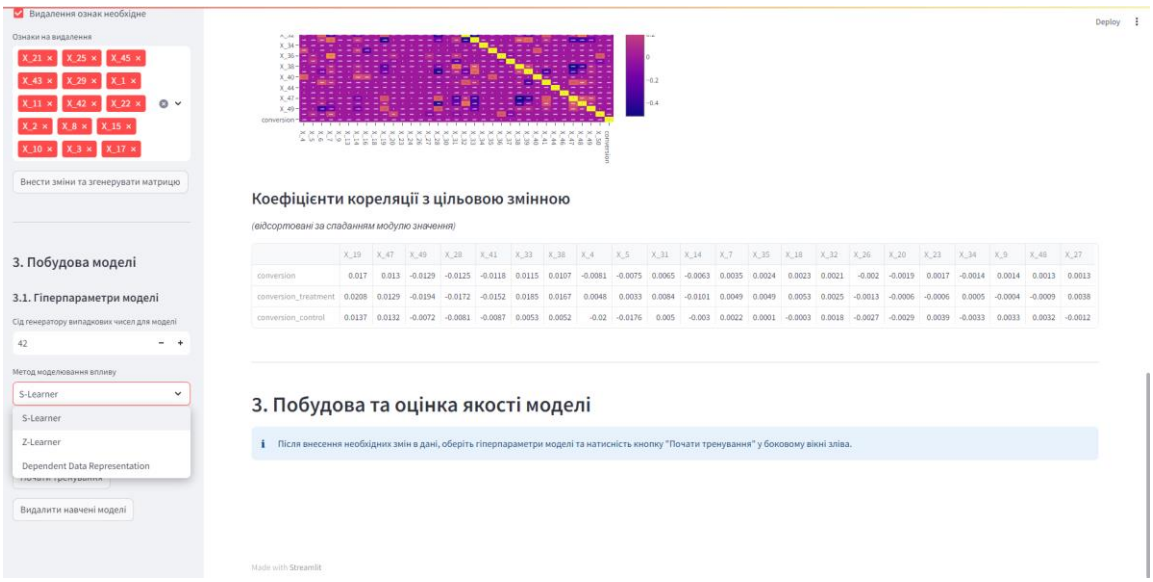


Рисунок 3.22. Можливість вибору виду методу моделювання впливу

Після натискання кнопки «Почати тренування» СППР тренує модель. Закінчивши тренувати модель, СППР повідомляє про успішне закінчення тренування моделі та пропонує оцінити якість моделі. В боковому вікні зліва можна вибрати моделі для порівняння та налаштувати параметри метрик оцінки їх якості (рис. 3.23).

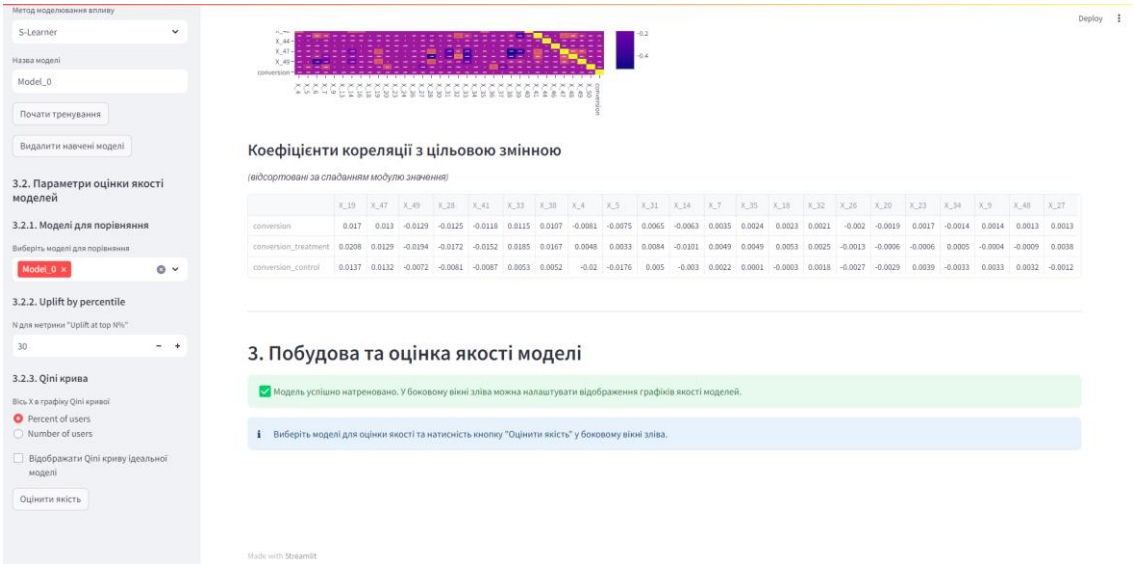


Рис. 3.23. Повідомлення про успішне закінчення тренування моделі

Після натискання кнопки «Оцінити якість» СППР виводить метрики та графіки якості першої моделі (рис. 3.24).

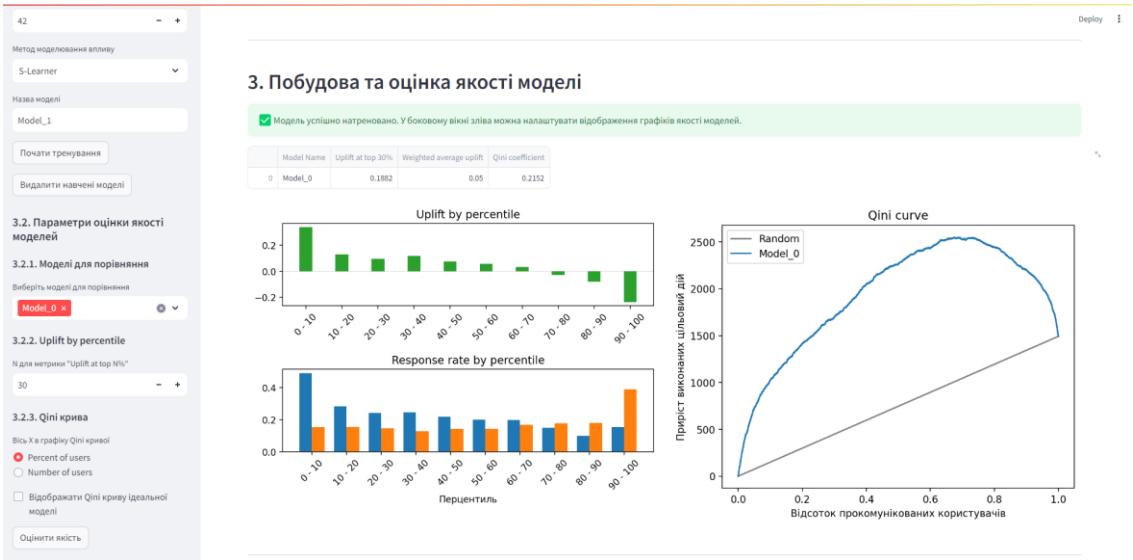


Рис. 3.24. Метрики та графіки якості першої моделі

Після тренування моделей на основі інших методів моделювання впливу та натискання кнопки «Оцінити якість» СППР виводить порівняння метрик та графіків якості всіх натренованих моделей (рис. 3.25).

Рис. 3.25. Порівняння метрик та графіків якості всіх натренованих моделей

Нижче СППР робить доступним новий розділ з можливістю завантаження нових користувачів та вибору натренованої моделі для оцінки впливу на них комунікації (рис. 3.26).

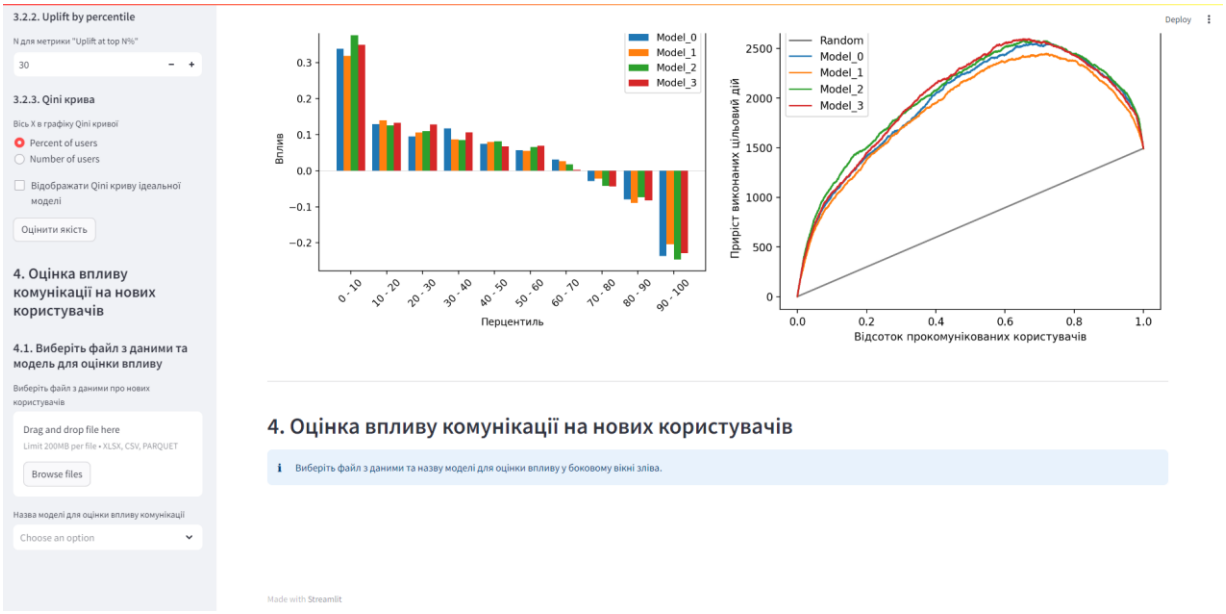


Рисунок 3.26. Розділ оцінки впливу комунікації на нових користувачів

Після вибору моделі та файлу з новими користувачами СППР розраховує та відображає оцінки впливу акції на них (рис. 3.27). Також є можливість вивантажити список ідентифікаторів нових користувачів з відповідними оцінками впливу у вигляді .csv файлу. Вміст такого файлу відображено на рисунку 3.28.

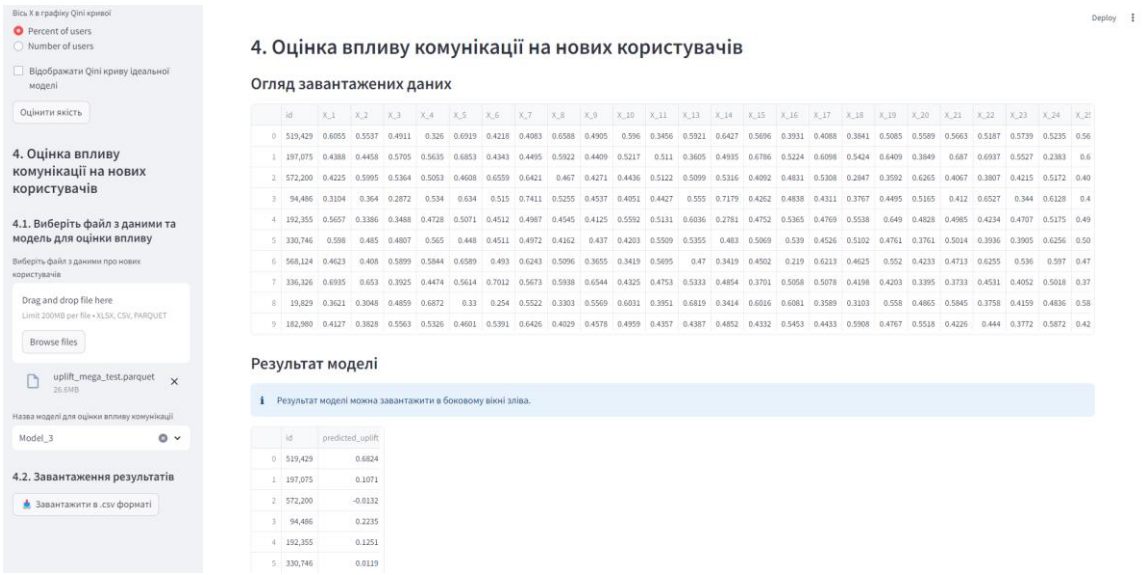
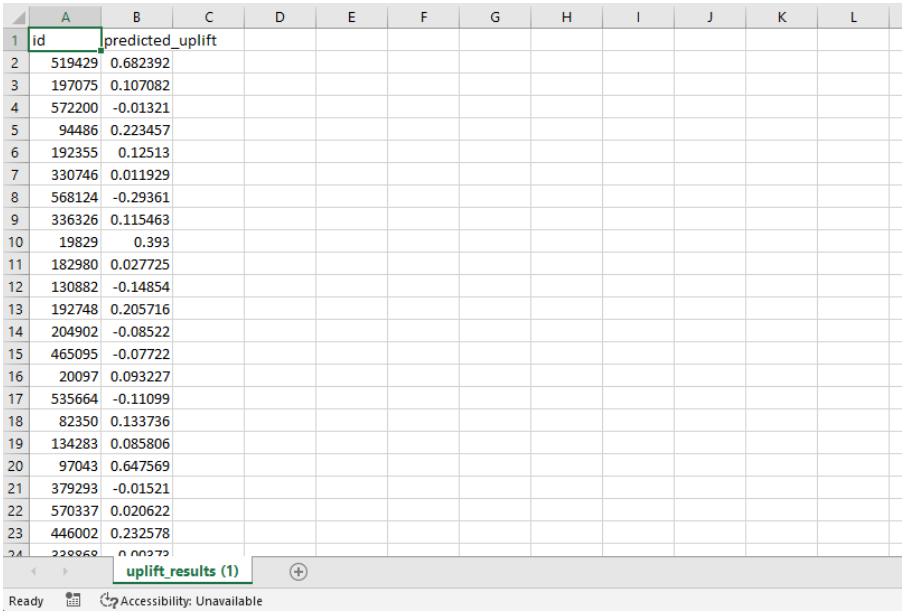


Рисунок 3.27. Результат оцінки впливу комунікації на нових користувачів



	A	B	C	D	E	F	G	H	I	J	K	L
1	id	predicted_uplift										
2	519429	0.682392										
3	197075	0.107082										
4	572200	-0.01321										
5	94486	0.223457										
6	192355	0.12513										
7	330746	0.011929										
8	568124	-0.29361										
9	336326	0.115463										
10	19829	0.393										
11	182980	0.027725										
12	130882	-0.14854										
13	192748	0.205716										
14	204902	-0.08522										
15	465095	-0.07722										
16	20097	0.093227										
17	535664	-0.11099										
18	82350	0.133736										
19	134283	0.085806										
20	97043	0.647569										
21	379293	-0.01521										
22	570337	0.020622										
23	446002	0.232578										
24	338868	0.00373										

Рисунок 3.28. Файл з оцінками впливу комунікації на нових користувачів

За наведеним прикладом можна побачити, що інтерфейс є інтуїтивно зрозумілим і СППР відповідає висунутим до неї вимогам.

3.2.3 Висновки реалізації СППР

В рамках розділу висунуто вимоги до СППР та обрано інструменти для створення. Приклад роботи з СППР та огляд інтерфейсу показують, що СППР відповідає висунутим до неї вимогам.

3.3 Висновки до розділу 3

В рамках розділу було проведено порівняльний аналіз DDR методу моделювання впливу з іншими методами та розроблено СППР на їх основі.

Порівняльний аналіз показав, що обидва варіанти обраного методу DDR показали кращі результати за інші розглянуті методи. Опираючись на результати, можна прийти до висновку, що якщо бюджет рекламної кампанії розрахований до 30% користувачів, то для розглянутого набору даних краще використовувати метод DDR з використанням результатів моделі контрольної групи в моделі тестової групи. У всіх інших випадках, краще використовувати метод DDR з використанням результатів моделі тестової групи в моделі контрольної групи для розглянутого набору даних.

Під час розробки СППР було висунуто вимоги до неї та обрано інструменти для створення. Приклад роботи з СППР та огляд інтерфейсу показують, що СППР відповідає висунутим до неї вимогам.

РОЗДІЛ 4 РОЗРОБКА СТАРТАП ПРОЄКТУ

Розділ має на меті проведення маркетингового аналізу стартап проекту задля визначення принципової можливості його ринкового впровадження та можливих напрямів реалізації цього впровадження. Система підтримки прийняття рішень з моделюванням впливу є актуальною і важливою у сучасному бізнесі з ряду причин.

1. Оптимізація маркетингових зусиль: моделювання впливу дозволяє компаніям ідентифікувати тих клієнтів, які мають потенціал позитивно відгукнутися на маркетингові акції, і, відповідно, зосереджувати свої ресурси на цільових групах. Це допомагає підвищити ефективність рекламних кампаній.

2. Зменшення втрат: за допомогою системи підтримки прийняття рішень з моделюванням впливу, компанії можуть уникнути надмірної розсилки пропозицій клієнтам, які вже готові придбати продукт без додаткового стимулювання, та клієнтам, які розсилка навпаки відштовхне від виконання цільової дії. Це допомагає зменшити витрати на маркетинг та збільшити прибуток.

3. Покращення взаємодії з клієнтами: завдяки здатності ідентифікувати "схильних до впливу" клієнтів, компанії можуть створювати персоналізовані пропозиції та комунікацію, що підвищує рівень задоволеності клієнтів та їх лояльність.

4. Адаптація до змін: сучасні ринки постійно змінюються, і попит клієнтів може коливатися. Моделювання впливу дозволяє компаніям швидко адаптувати свої маркетингові стратегії до нових умов і реагувати на зміни в клієнтському поведінці.

5. Новизна: моделювання впливу є доволі молодим та доволі маловідомим напрямком моделювання, який одночасно має великий потенціал.

Вищезазначене означає, що стартап є актуальним і має потенціал стати успішним на ринку.

4.1 План розробки стартапу та масштабування його на ринок

План розробки стартапу та його виведення на ринок складається з чотирьох етапів. Перший етап – це етап маркетингову аналізу стартапу, в рамках якого необхідно:

1) сформувати опис самої ідеї проекту та встановити загальні напрями використання майбутнього продукту, а також його відмінності від конкурентів;

2) провести аналіз ринкових можливостей для реалізації ідеї проекту;

3) на основі аналізу ринкового середовища розробити стратегію впровадження потенційного продукту на ринок.

Другим етапом є організація стартап проекту, в рамках якого необхідно:

1) підготувати календарний план і графік впровадження стартап-проекту;

2) розрахувати потребу в основних засобах і нематеріальних активах;

3) визначити плановий обсяг виробництва майбутнього продукту;

4) розрахувати загальні початкові витрати на запуск проекту та планові загальногосподарські витрати, необхідні для виконання проекту.

Третій етап це фінансово-економічний аналіз та оцінка ризиків проекту, в рамках якого необхідно:

1) визначити обсяг інвестиційних витрат;

2) провести розрахунки основних фінансово-економічних показників проекту, таких як обсяг виробництва, собівартість виробництва, ціна реалізації, податковий обсяг та чистий прибуток;

3) визначити рівень ризиків проекту і розробити стратегії їх запобігання та управління.

Останній етап це комерціалізація проекту. Він спрямований на пошук інвесторів та просування інвестиційної пропозиції (оферти) і включає:

1) визначення цільової групи інвесторів та опис їх ділових інтересів;

2) складання інвестиційної пропозиції (оферти), яка містить короткий огляд проекту для попереднього ознайомлення інвестора;

3) планування заходів з просування оферти, включаючи визначення комунікаційних каналів та площадок, а також планування ресурсів для реалізації цих заходів.

Виконання цих етапів у відповідній послідовності та у встановлені терміни створює передумови для успішного виходу на ринок [21].

4.2 Опис ідеї стартап-проекту

Стартап-проект полягає у вирішенні проблеми комунікації зі збитковими користувачами. Суть продукту стартапу полягає у розробці СППР для створення моделей, які допомагатимуть оцінити вплив комунікації на виконання цільової дії користувачем. З інформаційною картою стартапу можна ознайомитись в таблиці 4.1.

Таблиця 4.1 – Інформаційна карта стартап-проекту

Назва проекту	UpliftProphet
Автори проекту	Заіка Богдан Юрійович
Коротка анотація	Надати бізнесу та аналітикам програмний продукт для аналізу та оптимізації рекламних кампаній.
Термін реалізації проекту	12 місяців
Необхідні ресурси	Фінансові ресурси на оплату заробітної плати виконавцям протягом 12 місяців, а також на покриття оренди приміщення, комунальних послуги, техніка для виконавців, збереження даних, кіберзахист тощо.
Опис проблеми, яку вирішує проект	Продукт вирішує задачу визначення найвигідніших користувачів для проведення комунікації.
Головні цілі та завдання проекту	Метою проекту є надати бізнесу та аналітикам інструмент для оцінки впливу комунікації на виконання користувачами цільової дії
Очікувані результати	Привернення організацій з клієнтами до нашого стартапу для вирішення їх проблем з оптимізацією рекламних кампаній

4.3 Технологічний аудит ідеї проекту

З описом ідеї стартапу можна ознайомитись у таблиці 4.2.

Таблиця 4.2 – Опис ідеї стартапу

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Основна ідея полягає в створенні СППР, яка надасть бізнесу та аналітикам інструменти для оцінки впливу комунікації на виконання користувачем цільової дії	Опрацювання даних, створення моделі впливу та оцінка впливу комунікації на нову групу користувачів	Комунікація відбувається тільки з користувачами, для яких вона збільшить ймовірність виконання цільової дії, відповідно збільшується прибуток.
	Система збору відгуків про якість роботи програмного продукту	Відгуки дозволяються визначити напрямки покращення, покращення продукту збільшує задоволеність ним користувача

Порівняльний аналіз конкурентів знаходиться у таблиці 4.3.

Таблиця 4.3 – Порівняльний аналіз конкурентів проекту

№	Техніко-економічні ознаки ідеї	Товари/концепції конкурентів			W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	mParicle	Dataiku			
1	Зручність	5	2	4	-	-	+

Продовження таблиці 4.3

2	Простота	5	2	3	-	-	+
3	Повнота	4	4	5	-	+	-
4	Сумісність	5	3	4	-	-	+

Аналіз технічної здійсненності ідеї проекту описано в таблиці 4.4.

Таблиця 4.4 – Технологічна здійсненність продукту

№ п/п	Ідея проекту	Технології і реалізації	Наявність технологій	Доступність технологій
1	Створення СППР, яка надасть бізнесу та аналітикам інструменти для оцінки впливу комунікації на виконання користувачем цільової дії	Можливість завантаження власних даних з файлів	Наявні	Доступні
2		Можливість вибору різних методів попереднього аналізу даних	Наявні, необхідні допрацювання	Доступні
3		Можливість вибору різних методів моделювання впливу	Не наявні, необхідні допрацювання	Доступні
Обрана технологія реалізації ідеї проекту: Python, Streamlit				

4.4 Аналіз ринкових можливостей запуску стартап-проекту

Попередній аналіз ринку для запуску стартап-проекту знаходиться в таблиці 4.5.

Таблиця 4.5 – Характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	2
2	Загальний обсяг продаж, грн/ум.од	5000
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Професійна репутація
5	Специфічні вимоги до стандартизації та сертифікації	Відсутні
6	Середня норма рентабельності в галузі (або по ринку), %	15

Характеристику потенційних клієнтів, які можуть бути зацікавлені в проекті, описано в таблиці 4.6.

Таблиця 4.6 – Характеристика потенційних клієнтів стартап-проекту

№	Потреби, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Потреба в визначенні користувачів, комунікація з яким принесе найбільшу користь	Бізнес, який має велику базу клієнтів і потребує персоналізованої реклами	Клієнти з більшим бюджетом надаватимуть перевагу моделі, яка дозволяє максимізувати прибуток на всьому наборі клієнтів, в той час як клієнти з меншим бюджетом фокусуватимуться на моделях, які дають найбільший прибуток на топ Н % (де Н залежить від розміру бюджету)	- повинен бути інтуїтивно зрозумілим - повинен бути інтуїтивно зручним - має бути можливість використати створену модель на нових даних

Проведено аналіз загроз для розуміння можливих перешкод при запуску продукту на ринок (таблиця 4.7) та обраховано фактори можливостей для визначення сприятливих умов, щоб при можливості скористатись ними (таблиця 4.8).

Таблиця 4.7 – Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Конкуренція	Хоча ринок є відкритим і неосвоєним, на ньому вже є кілька великих гравців, які вже мають свою цільову групу покупців	Знайти точки додаткової цінності для користувача
2	Ціна збуту	Конкуренти можуть коштувати більше через те що їх бренд вже має репутацію.	На початку пропонувати більш вигідну умову для залучення нових користувачів ринку до себе та можливо перехоплення клієнтів конкурентів більш вигідними умовами
3	Якість аналізу	Через комплексність задачі, потрібно регулярно оновлювати моделі та методи моделювання впливу	Мати достатній штаб і ресурси, для аналізу сучасних проблем та швидкого оновлення

Таблиця 4.8 – Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Універсальність	Продукт не залежить від апаратної платформи	Зробити акцент на цьому при маркетингу, продовжувати розвиток як окремого продукту
2	Простота у використанні	Користувач без досвіду має з легкістю користуватись продуктом	Реалізувати інтуїтивно зрозумілий інтерфейс, який підказує що очікується від користувача
3	Якість та гарантії	Надавати найбільш якісні послуги	Оновлювати список методів моделювання впливу новими

Визначені тип та рівень конкуренції описані в таблиці 4.9.

Таблиця 4.9 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	У чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції: недосконала конкуренція	Представлено мало хороших методології та експертів	Зробити максимальним збут застосунку

Продовження таблиці 4.9

2. За рівнем конкурентної боротьби: міжнародний	Наявні проекти, розроблені та можуть бути доступні у всьому світі	Розширити цільову аудиторію, розробити підхід на різних мовах, пріоритезуючи їх за популярністю
3. За галузевою ознакою: внутрішньогалузева	Можуть працювати з різними галузями	Покращити персоналізацію
4. Конкуренція за видами товарів: товарно-родова	Конкуренція з аналізами інших систем та експертів	Підтримувати та покращувати якість існуючих функцій
5. За характером конкурентних переваг: нецінова	Різні компанії пропонують різну якість	Розробляти якісніші алгоритми і моделі
6. За інтенсивністю: марочна	Вже представлені компанії із сильним брендом	Предметно створити комунікаційну стратегію для побудови свого бренду

Далі необхідно виконати аналіз конкуренції за моделлю 5 сил конкуренції Майкла Портера (таблиця 4.10).

Таблиця 4.10 – Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти у галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	Інші існуючі методології та продукти	Якість, ціни, кількість користувачів, капіталовкладення	Фактори сили постачальників	Контроль якості, порівняння цін	Сила бренду, якість, ціна, масштаби
Висновки	Конкуренція з невеликою інтенсивністю, а також підігрітий ринок	Можливості входження на ринок, нові потенційні конкуренти	Постачальники відсутні	Клієнти диктують умови роботи на ринку	Товаро-замінники відсутні

Маючи результати аналізу конкуренції (таблиця 4.10), характеристики ідеї стартап-проекту (таблиця 4.5), характеристики потенційних клієнтів і їх вимоги до продукту (таблиця 4.6) та фактори ринкового середовища (таблиці 4.7 і 4.8) було сформульовано та обґрунтовано перелік факторів конкурентоспроможності (таблиця 4.11).

Таблиця 4.11 – Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспро-можності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Універсальність	Продукт не залежить від апаратної платформи, на відміну від більшості конкурентів
2	Простота у використанні	Від користувача треба лише дотримуватися інструкцій, які СППР надає впродовж всього шляху моделювання впливу
3	Якість та гарантії	Надавати найбільш якісні послуги
4	Безкоштовний сервіс при MVP	Максимально швидко набрати базу своїх клієнтів та заявити про себе на ринку

Аналіз сильних та слабких сторін продукту описано в таблиці 4.12.

Таблиця 4.12 – Порівняльний аналіз сильних та слабких сторін системи

№	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів						
			-3	-2	-1	0	1	2	3
1	Універсальність	20					+		
2	Простота у використанні	16	+						
3	Якість та гарантії	10			+				
4	Безкоштовний сервіс при MVP	18			+				

SWOT-аналіз [21, 22] продукту знаходиться в таблиці 4.13.

Таблиця 4.13 – SWOT-аналіз стартап-проекту

Сильні сторони Універсальність Простота у використанні Якість та гарантії	Слабкі сторони Відсутність сильного бренду Не сформована база клієнтів Не підключені альтернативні канали маркетингу
Можливості Покращення системи Персоналізація	Загрози Нові системи та експерти Збут

Завдяки проведенню SWOT-аналізу вдалось визначити сильні та слабкі сторони, можливості та загрози, пов'язані з конкуренцією та плануванням стартап-проекту. Далі було спроектовано альтернативну ринкову поведінку для інтеграції стартап-проекту на ринок та приблизний час реалізації системного комплексу, з урахуванням потенційних проектів, що можуть бути виведені на ринок. Результати наведено у таблиці 4.14.

Таблиця 4.14 – Альтернативи ринкового впровадження стартап проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Вихід на ринок з нижчою якістю	70%	4 місяці
2	Пропонувати одразу платне використання	50%	6 місяців

У даному підрозділі був проведений детальний аналіз ринку та продукту. Також відповідно до результатів проведеного конкурентного аналізу, визначених факторів ринку та його сприятливість, описання ідеї та характеристик стартап-проекту, можна зробити висновок, що існують дуже сприятливі умови для виходу продукту на ринок.

4.5 Розроблення ринкової стратегії стартап-проекту

Для розробки ринкової стратегії продукту, у першу чергу, необхідно проаналізувати цільову аудиторію проекту (таблиця 4.15).

Таблиця 4.15 – Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачі в сприйняти продукт	Орієнтовний попит у межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Персональні користувачі	Висока	25%	Висока	Середня
2	Великі бізнеси	Середня	20%	Середня	Середня
3	Малі і середні бізнеси	Висока	25%	Низька	Висока
4	Держава	Низька	10%	Низька	Висока
Які цільові групи обрано: 1, 3					

Далі визначається базова стратегія розвитку продукту (таблиця 4.16).

Таблиця 4.16 – Визначення базової стратегії розвитку

№	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	1 та 3	Диференційованого маркетингу	Масштабування та максимізація	Оптимальних витрат

Для роботи в обраних сегментах ринку сформовано базову стратегію розвитку (таблиці 4.17, 4.18).

Таблиця 4.17 – Визначення базової стратегії конкурентної поведінки

Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
Ні	Основний напрямок це нові споживачі, але перемануювання існуючих у конкурентів завдяки більш вигідним пропозиціям також планується	Ні	Стратегія інновацій та лідерства за вартістю

Таблиця 4.18 – Визначення стратегії позиціонування

Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
Універсальність Простота у використанні Якість результатів	Оптимальних витрат	Універсальність Простота у використанні Якість та гарантії Безкоштовне використання при MVP	Система, з інтуїтивно зрозумілим інтерфейсом та можливістю повноцінного проведення моделювання впливу

4.6 Розроблення маркетингової програми стартап-проекту

Після проведеного комплексного аналізу було описано ключові переваги концепції потенційного товару (таблиця 4.19) та побудовано концепцію маркетингових комунікацій (таблиця 4.20).

Таблиця 4.19 – Ключові переваги концепції потенційного товару

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити
1	Якісність результатів	Метод, який визначає найвигідніших для комунікації користувачів	Постійне покращення та оновлення методів згідно з новими внутрішніми та зовнішніми відкриттями
2	Універсальність	СППР не залежить від апаратної платформи	Це дозволяє користуватись СППР будь-якому користувачу
3	Простий інтерфейс	СППР дуже проста у використанні	СППР із інтуїтивно зрозумілим інтерфейсом

Таблиця 4.20 – Концепція маркетингових комунікацій

№	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламно о повідомлення	Концепція рекламного звернення
1	Пошук спеціалізованих систем	Реклама у соціальних мережах (з використанням власних моделей впливу)	Точність Якість Універсальність	Повідомлення про те, що це якісна методологія яка є незалежною	Реклама на цільову аудиторію

Продовження таблиці 4.20

2	Пошук доступного та дешевого продукту	Рекламні банери в Інтернеті, форуми	Простота Безкоштовне використання MVP	Вселити довіру у бренд та продукт	Реклама у лідерів думок Вивіски в публічних місцях Реклама на цільову аудиторію

4.7 Висновки до розділу 4

Даний розділ був присвячений дослідженню стартап-проекту – СППР на основі методів моделювання впливу.

У рамках розділу було досліджено розробку стратегій виходу на ринок та маркетинг-стратегії для цього. Зокрема, даний ринок являється сприятливим з невеликою кількістю представлених компаній конкурентів. Оскільки вони дають лише частину функцій, а запропонована система є універсальною та доступною, то у стартап-проекту є всі шанси стати монополістами на ринку.

Також були опрацьовані сильні та слабкі сторони проекту, аналіз конкурентів, SWOT-аналіз та цільової аудиторії. На основі всіх досліджень був сформований концепт маркетингової стратегії для обраних цільових аудиторій.

ВИСНОВКИ

Під час виконання магістерської дисертації було здійснено огляд технічної літератури за темою роботи, доведено актуальність обраної теми та проведено аналіз теоретичного матеріалу щодо моделювання впливу.

Моделювання впливу має велику актуальність, оскільки дозволяє підвищити ефективність маркетингу, зробити комунікації з клієнтами більш персоналізованими та оптимізувати бізнес-процеси. Потенціал досліджень в цьому напрямку великий через відносну новизну підходів, актуальність теми та відсутність уніфікованої теоретичної бази.

Проведено порівняльний аналіз обраного методу моделювання з іншими, який показав, що для обраного набору даних обидва варіанти методу з залежним представленням даних виявились найкращими, в залежності від потреб підприємства.

Розроблено СППР для оптимізації рекламних кампаній підприємства на основі розглянутих методів, яка, судячи з прикладу роботи та інтерфейсу, відповідає висунутим до неї вимогам. Вибір мови програмування Python та пакету Streamlit для створення інтерфейсу аргументовано, а методи моделювання впливу реалізовані таким чином в СППР, що дозволить легко масштабувати набір ймовірнісних класифікаторів доступних для використання в побудові нових моделей впливу. Реалізована СППР може використовуватись для повноцінного та швидкого моделювання впливу.

В результаті проведено аналіз потенціалу стартап-проекту на основі розробленої СППР, який показав, що ідея є актуальною та має великий попит.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Jaskowski M., Jaroszewicz S. Uplift modeling for clinical trial data. In: ICML Workshop on Clinical Data Analysis. 2012. P. 79-95. URL: https://people.cs.pitt.edu/~milos/icml_clinicaldata_2012/Papers/Oral_Jaroszewitz_ICML_Clinical_2012.pdf

2. McCaffrey D. F. Generalized additive models. SIAM Review, 1992, 34.4. P. 675-678.

3. Radcliffe N. J., Surry P. D. Real-world uplift modelling with significance-based uplift trees. White Paper TR-2011-1, Stochastic Solutions, 2011, P. 1-33. URL: https://www.stochasticsolutions.com/pdf/sig-based-up-trees.pdf

4. Zhao Z., Harinen T. Uplift modeling for multiple treatments with cost optimization. In: 2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA). IEEE, 2019. P. 422-431. URL: https://arxiv.org/pdf/1908.05372

5. Bon M., Feutry C., Meftah S. An in-depth benchmark study of the CATE estimation problem: experimental framework, metrics and models Version. URL: https://www.sjscience.org/files/papers/809/CoScience _809.pdf

6. Grundhoefer M. D., BANK U.S. Raising the bar in cross-sell marketing with uplift modeling. In: Predictive analytics world conference. 2009. URL: https://www.predictiveanalyticsworld.com/presentations/dc/2009/Michael Grundhoefer_Case%20StudyUSBank.pdf

7. Radclifte N. J.; Simpson R. Identifying who can be saved and who will be driven away by retention activity. Journal of Telecommunications Management, 2008, 1.2.

8. Gubela R. et al. Conversion uplift in e-commerce: A systematic benchmark of modeling strategies. International Journal of Information Technology & Decision Making, 2019, 18.03, P. 747-791. URL: https://www.econstor.eu/bitstream/10419/230773/1/irtg1792dp2018-062.pdf

9. Lo V.S.Y. The true lift model: a novel data mining approach to response modeling in database marketing. ACM SIGKDD Explorations Newsletter, 2002, 4.2. P. 78-86. URL: https://www.kdd.org/exploration_files/lo.pdf

10. Read J. et al. Classifier chains for multi-label classification. Machine learning, 2011, 85. P. 333-359. URL: https://link.springer.com/content/pdf/10.1007/s10994-011-5256-5.pdf

11. Betlei A., Diemert E., Amini M.-R. Uplift prediction with dependent feature representation in imbalanced treatment and control conditions. In: Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13–16, 2018, Proceedings, Part V 25. Springer International Publishing, 2018. P. 47-57. URL: https://bitlater.github.io/files/iconip_paper.pdf

12. Diemert E. et al. A large scale benchmark for uplift modeling. In: KDD. 2018. URL: https://hal.science/hal-02515860/file/large-scale-benchmark.pdf

13. Devriendt F., Moldovan D., Verbeke W. A literature survey and experimental evaluation of the state-of-the-art in uplift modeling: A stepping stone toward the development of prescriptive analytics. Big data, 2018, 6.1. P. 13-41. URL: https://www.liebertpub.com/doi/pdf/10.1089/ big.2017.0104

14. Michel R., Schnakenburg I., Von Martens T. Targeting uplift: An introduction to net scores. Springer Nature, 2019.

15. Radcliffe N. J., Surry P. D. Real-world uplift modelling with significance-based uplift trees. White Paper TR-2011-1, Stochastic Solutions, 2011, P. 1-33.

16. Radcliffe N. Using control groups to target on predicted lift: Building and assessing uplift model. Direct Marketing Analytics Journal, 2007, P. 14-21.

17. Gutierrez P., Gerardy J.-Y. Causal inference and uplift modelling: A review of the literature. In: International conference on predictive applications and APIs. PMLR, 2017. P. 1-13.

18. Відкритий датасет Megafon: веб-сайт. URL: https://www.uplift-modeling.com/en/latest/api/datasets/fetch_megafon.html

19. De Winter J. C. F., Gosling S. D., Potter J. Comparing the Pearson and Spearman correlation coefficients across distributions and sample sizes: A tutorial using simulations and empirical data. Psychological methods, 2016, 21.3. P. 273. URL: https://www.researchgate.net/profile/Joost-De-Winter/publication/307902372_Comparing_the_pearson_and_spearman_correlation_coefficients_across_distributions_and_sample_sizes_A_tutorial_using_simulations_and_empirical_data/links/63baa5d5c3c99660ebdc3f60/Comparing-the-pearson-and-spearman-correlation-coefficients-across-distributions-and-sample-sizes-A-tutorial-using-simulations-and-empirical-data.pdf

20. Заіка Б. Ю., Терентьєв О. М. Система підтримки прийняття рішень для оптимізації рекламних кампаній підприємства на основі методу моделювання впливу з залежним представленням даних, II науково-практична конференція «Системні науки та інформатика», КПІ ім. Ігоря Сікорського, Київ, 4-8 грудня, 2023. С. 108-116.

21. Гавриша О. А. Розроблення стартап-проекту: Методичні рекомендації до виконання розділу магістерських дисертацій для студентів інженерних спеціальностей. Київ : НТУУ «КПІ», 2016. 28 с. URL: https://ela.kpi.ua/bitstream/123456789/35763/1/startap_proekt_MV.pdf

22. Згуровський М. З., Панкратова Н. Д. Основи системного аналізу: підруч. [для студ. вищ. навч. закл.] К. : Вид. група ВНУ, 2007. 544 с.

ДОДАТОК А КОД ПРОГРАМНОГО ПРОДУКТУ

application.py

```
from functions import *


if 'show_corr_map' not in st.session_state:
        clean_vars()

st.set_page_config(
    page_title='DSS for Uplift Modeling with DDR method',
    page_icon='🎓',
    layout='wide'
)

st.write(st.session_state)
st.write("# Decision Support System for Uplift Modeling with Dependent Data Representation method")
st.write('## 1. Огляд завантажених даних')

with st.sidebar:
    st.write('# 1. Завантаження даних')
    st.write('## 1.1. Виберіть спосіб розбиття даних на тренувальні та тестові')
    train_test_options = [
        'Завантажити один файл з даними та розбити їх на тренувальні та тестові в програмі',
        'Завантажити два окремих файли з тренувальними та тестувальними даними відповідно'
    ]
    split_method = st.radio('Спосіб розбиття даних на тренувальні та тестові', options=train_test_options, index=None)

if split_method == 'Завантажити один файл з даними та розбити їх на тренувальні та тестові в програмі':
    with st.sidebar:
```

```
        st.write('## 1.2. Виберіть файл з даними')
        data_file = st.file_uploader("Виберіть файл з даними", type=["xlsx", "csv",
"parquet"])
        train_file, test_file = None, None

    if data_file is not None:
        data = file_to_df(data_file)
        st.write('Розмір оригінальних даних:', data.shape)
        with st.sidebar:
            st.write('## 1.3. Виберіть значення параметрів для розбиття даних')
            split_rand_seed = st.number_input('Сід генератору випадкових чисел для
розбиття даних', min_value=1, value=42)
            split_test_size = st.number_input('Відсоток тренувальних даних',
min_value=0, max_value=100, value=20) / 100
        data_train, data_test = train_test_split(data, test_size=split_test_size,
random_state=split_rand_seed)

if split_method == 'Завантажити два окремих файли з тренувальними та
тестувальними даними відповідно':
    with st.sidebar:
        st.header('1.2. Виберіть файли з даними')
        data_file = None
        train_file = st.file_uploader("Виберіть файл з тренувальними даними",
type=["xlsx", "csv", "parquet"])
        test_file = st.file_uploader("Виберіть файл з тестовими даними",
type=["xlsx", "csv", "parquet"])

    if train_file is not None and test_file is not None:
        data_train = file_to_df(train_file)
        data_test = file_to_df(test_file)

if split_method is not None:
    if data_file is not None or (train_file is not None and test_file is not None):
        c1, c2 = st.columns(2)
        with c1:
            st.write('### Тренувальні дані')
            data_info(data_train)
        with c2:
```

```
            st.write('### Тестові дані')
            data_info(data_test)
        st.divider()


        st.write('## 2. Коефіцієнти рангової кореляції Спірмена в тренувальних даних')
        with st.sidebar:
            st.divider()
            st.write('# 2. Аналіз тренувальних даних')
            st.write('## 2.1. Виберіть які ознаки відображають:')
            id_col = st.selectbox('Ідентифікатор користувача', list(data_train.columns), index=None)

            if id_col is not None:
                treatment_col = st.selectbox('Ознака комунікації', list(data_train.drop(columns=[id_col]).columns), index=None)
                if treatment_col is not None:
                    response_col = st.selectbox('Ознака виконання цільової дії', list(data_train.drop(columns=[id_col, treatment_col]).columns), index=None)

        if id_col is not None and treatment_col is not None and response_col is not None:
            with st.sidebar:
                st.write('## 2.2. Видалення ознак та генерування кореляційної матриці')
                del_required = st.checkbox('Видалення ознак необхідне')
                if del_required:
                    ban_cols = st.multiselect('Ознаки на видалення', list(data_train.drop(columns=[id_col, treatment_col, response_col]).columns))
                        )
                else:
                    ban_cols = []

                generate_corr_map_clicked = st.button('Внести зміни та згенерувати матрицю')

            if generate_corr_map_clicked:
```

```
            cols = list(data_train.drop(columns=[id_col,
treatment_col]+ban_cols).columns)
            with st.spinner('Розрахунок коефіцієнтів кореляції в процесі...'):
                st.session_state.show_corr_map = True
                corr_comp =
data_train[cols].corr(method='spearman')[response_col].reset_index().merge(
                    data_train.loc[data_train[treatment_col]==0,
cols].astype(float).corr(method='spearman')[response_col].reset_index(),
on='index', suffixes=('', '_treatment')
                ).merge(
                    data_train.loc[data_train[treatment_col]==1,
cols].astype(float).corr(method='spearman')[response_col].reset_index(),
on='index', suffixes=('', '_control')
                ).sort_values(by=response_col, ascending=False, key = lambda x:
abs(x)).round(4).set_index('index').iloc[1:].T
                st.session_state.corr_response = corr_comp
                st.session_state.corr_map = corr_heatmap(data_train[cols])

        if st.session_state.show_corr_map:
            st.write('### Кореляційна матриця')
            st.plotly_chart(st.session_state.corr_map, theme=None)
            st.write('### Коефіцієнти кореляції з цільовою змінною')
            st.write('*(відсортовані за спаданням модулю значення)*')
            st.write(st.session_state.corr_response)
            st.divider()


            st.write('## 3. Побудова та оцінка якості моделі')
            X_train, y_train, t_train =
get_x_y_t(data_train.drop(columns=ban_cols), id_col, response_col,
treatment_col)
            X_test, y_test, t_test = get_x_y_t(data_test.drop(columns=ban_cols),
id_col, response_col, treatment_col)
            with st.sidebar:
                st.divider()
                st.write('# 3. Побудова моделі')
                st.write('## 3.1. Гіперпараметри моделі')
```

```
            model_rand_seed = st.number_input('Сід генератору випадкових
чисел для моделі', min_value=1, value=42)
            method_type = st.selectbox('Метод моделювання впливу', ['S-
Learner', 'Z-Learner', 'Dependent Data Representation'])
            if method_type == 'Dependent Data Representation':
                method_ddr_feature = st.selectbox('Група, результати моделі якої
будуть ознакою для моделі іншої групи', ['treatment', 'control'])
            model_name = st.text_input('Назва моделі',
f'Model_{st.session_state.model_counter}')
            train_model_clicked = st.button('Почати тренування')
            clean_models_clicked = st.button('Видалити навчені моделі')


        if train_model_clicked:
            with st.spinner('Тренування моделі в процесі...'):
                if method_type == 'Dependent Data Representation':
                    st.session_state.last_model = TwoModelsDDR(

classifier_trmt=XGBClassifier(random_state=model_rand_seed),

classifier_ctrl=XGBClassifier(random_state=model_rand_seed),
                        ddr_feature=method_ddr_feature
                    )
                elif method_type == 'Z-Learner':
                    st.session_state.last_model = ZLearner(
                        classifier=XGBClassifier(random_state=model_rand_seed)
                    )
                elif method_type == 'S-Learner':
                    st.session_state.last_model = SLearner(
                        classifier=XGBClassifier(random_state=model_rand_seed)
                    )
                st.session_state.last_model.fit(X_train, y_train, t_train)
                st.session_state.last_model_uplift =
st.session_state.last_model.predict(X_test)
                st.session_state.models[model_name] = st.session_state.last_model
                st.session_state.model_uplifts[model_name] =
st.session_state.last_model_uplift
                st.session_state.model_counter += 1
```

```
            if clean_models_clicked:
                clean_model_vars()
                clean_eval_vars()

            if st.session_state.last_model is not None:
                st.success('Модель успішно натреновано. У боковому вікні зліва можна налаштувати відображення графіків якості моделей.', icon="✅")
                with st.sidebar:
                    st.write('## 3.2. Параметри оцінки якості моделей')
                    st.write('### 3.2.1. Моделі для порівняння')
                    all_models_keys = list(st.session_state.models.keys())
                    def_models_keys = all_models_keys
                    if len(def_models_keys) > 4:
                        def_models_keys = def_models_keys[:4]
                    chosen_models_keys = st.multiselect(
                        'Виберіть моделі для порівняння', all_models_keys, default = def_models_keys, max_selections=4
                    )
                    st.write('### 3.2.2. Uplift by percentile')
                    split_test_size = st.number_input('N для метрики "Uplift at top N%"', min_value=0, max_value=100, value=30) / 100
                    st.write('### 3.2.3. Qini крива')
                    qini_x_type = st.radio('Вісь X в графіку Qini кривої', options=['Percent of users', 'Number of users']) == 'Percent of users'
                    perfect_qini_required = st.checkbox('Відображати Qini криву ідеальної моделі')
                    eval_models_clicked = st.button('Оцінити якість')

                if eval_models_clicked:
                    if len(chosen_models_keys) == 0:
                        st.error('Виберіть унікальну назву для моделі та повторно натисність кнопку "Почати тренування".')
                    else:
                        st.session_state.show_eval_metrics = True
                        st.session_state.ubp_fig, ubp_metrics = plot_uplift_by_percentile_barchart(t_test, y_test, chosen_models_keys, top_perc=split_test_size)
```

```
                    st.session_state.qc_fig, qc_metrics = plot_qini_curve(t_test,
y_test, chosen_models_keys, normalize_n=qini_x_type,
plot_perfect=perfect_qini_required)
                    st.session_state.eval_metrics = ubp_metrics.merge(qc_metrics,
on='Model Name')

                if st.session_state.show_eval_metrics:
                    st.write(st.session_state.eval_metrics)
                    c31, c32 = st.columns(2)
                    with c31:
                        st.pyplot(st.session_state.ubp_fig)
                    with c32:
                        st.pyplot(st.session_state.qc_fig)
                    st.divider()

                    st.write('## 4. Оцінка впливу комунікації на нових користувачів')
                    with st.sidebar:
                        st.write('# 4. Оцінка впливу комунікації на нових
користувачів')
                        st.write('## 4.1. Виберіть файл з даними та модель для оцінки
впливу')
                        new_users_file = st.file_uploader("Виберіть файл з даними про
нових користувачів", type=["xlsx", "csv", "parquet"])
                        predict_model_key = st.selectbox('Назва моделі для оцінки
впливу комунікації', all_models_keys, index=None)

                    if new_users_file is not None and predict_model_key is not None:
                        new_users = file_to_df(new_users_file)
                        st.write('### Огляд завантажених даних')
                        st.write(new_users.head(10))
                        with st.spinner('Розрахунок оцінок впливу комунікації в
процесі...'):
                            X_new_users = new_users.drop(columns=ban_cols+[id_col])
                            uplift_new_users =
st.session_state.models[predict_model_key].predict(X_new_users)
                            uplift_new_users = pd.concat([new_users[id_col],
pd.Series(uplift_new_users, name='predicted_uplift')], axis=1)
                        st.write('### Результат моделі')
```

```
                info_msg('Результат моделі можна завантажити в боковому
вікні зліва.')
                st.write(uplift_new_users)
                with st.sidebar:
                    st.write('## 4.2. Завантаження результатів')
                    st.download_button(
                        label="📥 Завантажити в .csv форматі",
                        data=uplift_new_users.to_csv(index=False),
                        file_name='uplift_results.csv',
                        mime='text/csv',
                    )


            else:
                info_msg('Виберіть файл з даними та назву моделі для оцінки
впливу у боковому вікні зліва.')


        else:
            info_msg('Виберіть моделі для оцінки якості та натисність
кнопку "Оцінити якість" у боковому вікні зліва.')
            clean_eval_vars()


    else:
        info_msg('Після внесення необхідних змін в дані, оберіть
гіперпараметри моделі та натисність кнопку "Почати тренування" у боковому
вікні зліва.')
        clean_model_vars()
        clean_eval_vars()


else:
    info_msg('Виберіть зайві ознаки на видалення (якщо такі присутні) у
боковому вікні зліва та натисніть кнопку "Внести зміни та згенерувати
матрицю", щоб видалити обрані ознаки та порахувати кореляційні
коефіцієнти.')
    clean_vars()
```

```
        else:
            info_msg('Виберіть ознаки, які відображають ідентифікатор користувача, ознаку комунікації та ознаку виконання цільової дії, у боковому вікні зліва щоб продовжити.')
            clean_vars()


    else:
        info_msg('Виберіть всі файли з даними у боковому вікні зліва щоб продовжити.')
        clean_vars()


if split_method is None:
    info_msg('Виберіть спосіб розбиття даних у боковому вікні зліва щоб продовжити.')
    clean_vars()
```

functions.py

```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
from sklearn.base import BaseEstimator, clone
from sklearn.utils.validation import check_X_y, check_array, check_is_fitted
from xgboost import XGBClassifier

import streamlit as st
import plotly.express as px
```

```
# >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>> app functions
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
def info_msg(text):
    st.info(text, icon="ℹ")

def clean_corr_vars():
    st.session_state.show_corr_map = None
    st.session_state.corr_map = None
    st.session_state.corr_response = None

def clean_model_vars():
    st.session_state.model_counter = 0
    st.session_state.last_model = None
    st.session_state.last_model_uplift = None
    st.session_state.models = dict()
    st.session_state.model_uplifts = dict()

def clean_eval_vars():
    st.session_state.show_eval_metrics = None
    st.session_state.ubp_fig = None
    st.session_state.qc_fig = None
    st.session_state.eval_metrics = None

def clean_vars():
    clean_corr_vars()
    clean_model_vars()
    clean_eval_vars()

def file_to_df(data_file):
    if data_file.name.split('.')[1] == 'xlsx':
        return pd.read_excel(data_file)
    elif data_file.name.split('.')[1] == 'parquet':
        return pd.read_parquet(data_file)
    elif data_file.name.split('.')[1] == 'csv':
        return pd.read_csv(data_file)

def data_info(data):
    st.write('Розмір даних:', data.shape)
```

```
    st.write('Даних:', data.shape)
    st.dataframe(data.head(10))
    dtypes = pd.DataFrame(data.dtypes).T
    dtypes.index = ['dtype']
    st.write('Статистичні характеристики даних:')
    st.dataframe(pd.concat([dtypes,
data.describe(include='all').round(4)]).astype(str))

def corr_heatmap(data):
    corr = data.corr(method='spearman').round(3)
    fig = px.imshow(corr, text_auto=True, aspect="auto")
    return fig

def get_x_y_t(data, id_col, y_col, t_col):
    df = data.drop(columns=id_col)
    return df.drop(columns=[t_col, y_col]), df[y_col], df[t_col]


# >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>> uplift functions
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

# >>>>> Model
class SLearner(BaseEstimator):
    def __init__(self, classifier):
        self.classifier = clone(classifier)


    def fit(self, X, y, t):
        X, y = check_X_y(X, y)
        t = check_array(t, ensure_2d=False)

        X_t = np.concatenate((X, t.reshape(-1, 1)), axis=1)
        self.classifier.fit(X_t, y)

        return self


    def predict(self, X):
```

```
        check_is_fitted(self.classifier)
        X = check_array(X)
        X_1 = np.concatenate((X, np.ones((X.shape[0], 1))), axis=1)
        X_0 = np.concatenate((X, np.zeros((X.shape[0], 1))), axis=1)

        return  self.classifier.predict_proba(X_1)[:, 1] -
self.classifier.predict_proba(X_0)[:, 1]


class ZLearner(BaseEstimator):
    def __init__(self, classifier):
        self.classifier = clone(classifier)


    def fit(self, X, y, t):
        X, y = check_X_y(X, y)
        t = check_array(t, ensure_2d=False)

        z = y * t + (1 - y) * (1 - t)
        self.classifier.fit(X, z)

        return self


    def predict(self, X):
        check_is_fitted(self.classifier)
        X = check_array(X)

        return  2 * self.classifier.predict_proba(X)[:, 1] - 1


class TwoModelsDDR(BaseEstimator):
    def __init__(self, classifier_trmt, classifier_ctrl, ddr_feature='treatment'):
        self.classifier_trmt = clone(classifier_trmt)
        self.classifier_ctrl = clone(classifier_ctrl)
        if ddr_feature in ('treatment', 'control'):
            self.ddr_feature = ddr_feature
        else:
```

```
            raise TypeError("Only 'treatment' or 'control' values are allowed in
ddr_feature")


    def fit(self, X, y, t):
        X, y = check_X_y(X, y)
        t = check_array(t, ensure_2d=False)

        X_trmt = np.copy(X[t==1])
        y_trmt = np.copy(y[t==1])
        X_ctrl = np.copy(X[t==0])
        y_ctrl = np.copy(y[t==0])

        if self.ddr_feature == 'treatment':
            self.classifier_trmt.fit(X_trmt, y_trmt)

            trmt_prob = self.classifier_trmt.predict_proba(X_ctrl)[:, 1].reshape(-1, 1)
            X_ctrl = np.concatenate((X_ctrl, trmt_prob), axis=1)
            self.classifier_ctrl.fit(X_ctrl, y_ctrl)

        elif self.ddr_feature == 'control':
            self.classifier_ctrl.fit(X_ctrl, y_ctrl)

            ctrl_prob = self.classifier_ctrl.predict_proba(X_trmt)[:, 1].reshape(-1, 1)
            X_trmt = np.concatenate((X_trmt, ctrl_prob), axis=1)
            self.classifier_trmt.fit(X_trmt, y_trmt)

        else:
            raise TypeError("Only 'treatment' or 'control' values are allowed in
ddr_feature")

        return self


    def predict(self, X):
        check_is_fitted(self.classifier_trmt)
        check_is_fitted(self.classifier_ctrl)
        X = check_array(X)
```

```
        if self.ddr_feature == 'treatment':
            trmt_prob = self.classifier_trmt.predict_proba(X)[:, 1]
            X = np.concatenate((X, trmt_prob.reshape(-1, 1)), axis=1)
            ctrl_prob = self.classifier_ctrl.predict_proba(X)[:, 1]

        elif self.ddr_feature == 'control':
            ctrl_prob = self.classifier_ctrl.predict_proba(X)[:, 1]
            X = np.concatenate((X, ctrl_prob.reshape(-1, 1)), axis=1)
            trmt_prob = self.classifier_trmt.predict_proba(X)[:, 1]
        else:
            raise TypeError("Only 'treatment' or 'control' values are allowed in
ddr_feature")

        return trmt_prob - ctrl_prob


# >>>>> Metrics
def uplift_at_top_perc(treatment_flg, response, uplift, top_perc=0.3):
    df = pd.DataFrame([[val1, val2, val3] for val1, val2, val3 in
zip(list(treatment_flg), list(response), list(uplift))], columns=['treatment_flg',
'response', 'uplift'])
    df = df.sort_values(by='uplift', ascending=False)
    df = df.iloc[:int(df.shape[0] * top_perc), :].copy()

    return df[df['treatment_flg']==1]['response'].mean() -
df[df['treatment_flg']==0]['response'].mean()

def uplift_by_percentile_table(treatment_flg, response, uplift):
    tmp = pd.DataFrame([[val1, val2, val3] for val1, val2, val3 in
zip(list(treatment_flg), list(response), list(uplift))], columns=['treatment_flg',
'response', 'uplift'])
    tmp['percentile'] = pd.qcut(tmp['uplift'], 10, labels=[f'{i*10} - {(i+1)*10}' for i
in range(9, -1, -1)])

    tmp2 = tmp.groupby(['percentile', 'treatment_flg']).agg(
        n = ('response', 'count'),
        avg_treatment = ('response', 'mean')
```

```
    ).reset_index()

    ubp_table = tmp2.pivot(index='percentile', columns='treatment_flg')[['n',
'avg_treatment']]
    ubp_table.columns = [f'{a}_{b}' for a in ['n', 'response_rate'] for b in ['control',
'treatment']]
    ubp_table = ubp_table[[f'{a}_{b}' for a in ['n', 'response_rate'] for b in
['treatment', 'control']]].copy()
    ubp_table = ubp_table.reset_index().sort_values(by='percentile',
ascending=False)
    ubp_table['uplift'] = ubp_table['response_rate_treatment'] -
ubp_table['response_rate_control']

    return ubp_table

def plot_uplift_by_percentile_barchart(treatment_flg, response, model_keys,
top_perc=0.3):
    if len(model_keys) == 1:
        ubp_table = uplift_by_percentile_table(treatment_flg, response,
st.session_state.model_uplifts[model_keys[0]])

        uatp = uplift_at_top_perc(treatment_flg, response,
st.session_state.model_uplifts[model_keys[0]], top_perc=top_perc)
        wau = (ubp_table['n_treatment'] * ubp_table['uplift']).sum() /
ubp_table['n_treatment'].sum()
        ubp_metrics = pd.DataFrame({
            'Model Name': [model_keys[0]],
            f'Uplift at top {top_perc:.0%}': [uatp],
            'Weighted average uplift': [wau]
        })

        fig, (ax1, ax2) = plt.subplots(nrows=2, ncols=1)
        fig.set_tight_layout(True)
        width = 0.36
        r = np.arange(ubp_table.shape[0])

        ax1.set_title(f'Uplift by percentile')
        ax1.bar(r, ubp_table['uplift'], color = 'tab:green',
```

```
            width = width, label='Uplift')
        ax1.axhline(color='black', linewidth=.1)

        ax2.set_title('Response rate by percentile')
        ax2.bar(r-width/2, ubp_table['response_rate_treatment'], color = 'tab:blue',
            width = width, label='Treatment response rate')
        ax2.bar(r+width/2, ubp_table['response_rate_control'], color = 'tab:orange',
            width = width, label='Control response rate')

        for ax in (ax1, ax2):
            ax.set_xticks(r, ubp_table['percentile'], rotation=45)

        plt.xlabel('Перцентиль')

        return fig, ubp_metrics

    else:
        ubp_metrics = pd.DataFrame()

        fig, ax = plt.subplots(nrows=1, ncols=1)
        fig.set_tight_layout(True)
        ax.set_title(f'Uplifts by percentile')

        width = 0.2
        r = np.arange(10)
        n = len(model_keys)
        colors = [f'tab:{color}' for color in ['blue', 'orange', 'green', 'red']]


        for i in range(n):
            ubp_table = uplift_by_percentile_table(treatment_flg, response,
st.session_state.model_uplifts[model_keys[i]])

            uatp = uplift_at_top_perc(treatment_flg, response,
st.session_state.model_uplifts[model_keys[i]], top_perc=top_perc)
            wau = (ubp_table['n_treatment'] * ubp_table['uplift']).sum() /
ubp_table['n_treatment'].sum()
            ubp_metrics = pd.concat([ubp_metrics, pd.DataFrame({
```

```
                'Model Name': [model_keys[i]],
                f'Uplift at top {top_perc:.0%}': [uatp],
                'Weighted average uplift': [wau]
            })], axis=0)

        ax.bar(r - width*(n-1)/2 + i*width, ubp_table['uplift'], color = colors[i],
            width = width, label = model_keys[i])


    ax.axhline(color='black', linewidth=.1)
    ax.set_xticks(r, ubp_table['percentile'], rotation=45)
    plt.xlabel('Перцентиль')
    plt.ylabel('Вплив')
    plt.legend()

    return fig, ubp_metrics



def calc_qini_curve(treatment_flg, response, uplift, normalize_n=True,
return_uplift=False):
    df = pd.DataFrame([[val1, val2, val3] for val1, val2, val3 in
zip(list(treatment_flg.astype(bool)), list(response), list(uplift))],
columns=['treatment_flg', 'response', 'uplift'])
    df = df.sort_values(by='uplift', ascending=False)

    df['y_t'] = (df['response'] & df['treatment_flg']).astype(int).cumsum()
    df['y_c'] = (df['response'] & ~df['treatment_flg']).astype(int).cumsum()
    df['n_t'] = (df['treatment_flg']).astype(int).cumsum()
    df['n_c'] = (~df['treatment_flg']).astype(int).cumsum()

    df['qini'] = df.apply(lambda row: row['y_t'] - row['y_c'] * row['n_t'] /
max(row['n_c'], 1), axis=1)

    df = df.reset_index(drop=True)
    df.index.name = 'n'
    df = df.reset_index()
    df['n'] = df['n'] + 1
```

```
    df = pd.concat([pd.DataFrame([[0, 0]], columns=['n', 'qini']), df])

    if normalize_n:
        df['n'] = df['n'] / df['n'].max()

    if return_uplift:
        return df[['n', 'qini', 'uplift']]

    return df[['n', 'qini']]

def calc_perfect_qini_curve(treatment_flg, response, normalize_n=True,
return_uplift=False):
    return calc_qini_curve(treatment_flg, response, response * treatment_flg -
response * (1 - treatment_flg), normalize_n=normalize_n,
return_uplift=return_uplift)

def plot_qini_curve(treatment_flg, response, model_keys, normalize_n=True,
plot_perfect=False):
    perfect_curve = calc_perfect_qini_curve(treatment_flg, response,
normalize_n=normalize_n)
    random_curve = perfect_curve.iloc[[0, -1], :]

    perfect_auc = np.trapz(perfect_curve['qini'], perfect_curve['n'])
    random_auc = np.trapz(random_curve['qini'], random_curve['n'])

    qc_metrics = pd.DataFrame()

    fig, ax = plt.subplots()
    ax.set_title(f'Qini curve')
    ax.plot(random_curve['n'], random_curve['qini'], label='Random',
color='tab:gray')
    if plot_perfect:
        ax.plot(perfect_curve['n'], perfect_curve['qini'], label='Perfect',
color='tab:purple')



    colors = [f'tab:{color}' for color in ['blue', 'orange', 'green', 'red']]
```

```
    for i in range(len(model_keys)):
        model_curve = calc_qini_curve(treatment_flg, response,
st.session_state.model_uplifts[model_keys[i]], normalize_n=normalize_n)
        model_auc = np.trapz(model_curve['qini'], model_curve['n'])
        qc = (model_auc - random_auc) / (perfect_auc - random_auc)
        qc_metrics = pd.concat([qc_metrics, pd.DataFrame({
                'Model Name': [model_keys[i]],
                f'Qini coefficient': [qc]
            })], axis=0)
        ax.plot(model_curve['n'], model_curve['qini'], label=model_keys[i],
color=colors[i])

    if normalize_n:
        ax.set_xlabel('Відсоток прокомунікованих користувачів')
    else:
        ax.set_xlabel('Кількість прокомунікованих користувачів')
    ax.set_ylabel('Приріст виконаних цільовий дій')
    ax.legend()

    return fig, qc_metrics
```