МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Харківський національний університет імені В.Н.Каразіна

Факультет математики і інформатики

Кафедра теоретичної та прикладної інформатики

Кваліфікаційна робота

бакалавр

на тему «Розробка інтегрованої програмної системи аналітики для Інтернет-маркетингу: дослідження алгоритмів прогнозування та можливостей візуалізації даних»

Виконав: студент 4 курсу, групи МФ-41

спеціальність 122 «Комп'ютерні науки»

освітньо-професійна програма «Теоретична і прикладна інформатика»

Безуглий Олександр Олегович

(прізвище та ініціали)

Керівник Меняйлов Є.С

(прізвище та ініціали)

Рецензент_____

(прізвище та ініціали)

Харків – 2024 року

ЗАТВЕРДЖУЮ

В.о. зав. кафедри

теоретичної та прикладної інформатики

Меняйлов Є. С.

_____

підпис ініціали, прізвище

“____” _____20___року

З А В Д А Н Н Я

НА ДИПЛОМНУ РОБОТУ

Безуглому Олександру Олегович

(прізвище, ім’я, по батькові студента)

1. Тема роботи Розробка інтегрованої програмної системи аналітики для Інтернет-маркетингу: дослідження алгоритмів прогнозування та можливостей візуалізації даних»

керівник роботи Меняйлов Євген Сергійович, кандидат технічних наук, доцент _____,

(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “___”_____20__року №___

2. Строк подання студентом роботи_____

3. Перелік питань, які потрібно розробити)

1. Аналіз предметної області. 1.1 Аналіз існуючих систем та методів аналізу маркетингових даних 1.2 Показники Інтернет- маркетингу 1.3Аналіз BI систем для візуалізації даних 1.4 Постановка задачі 2 Математична модель формування прогнозних оцінок 2.1 Оцінка точності моделі. 2.2 Обґрунтування вибору методу sBG 3 Структурний аналіз програмної системи 4 Розробка та аналіз моделі даних 5 Розробка алгоритму формування прогнозних оцінок доходу 6. Візуалізація та аналіз отриманих результатів

4. План роботи

№	Назва етапів роботи
1.	*Отримання завдання кваліфікаційної роботи*
2.	*Аналіз завдання, літератури та аналогів з теми кваліфікаційної роботи*
3.	*Аналіз та дослідження математичних моделей прогнозування*
4.	*Дослідження структури розроблювальної задачі методами структурного моделювання*
5.	*Розробка структури та аналізу моделі даних*
6.	*Дослідження методів та інструментарію візуалізації даних*
7.	*Розроблення алгоритму моделі прогнозування*
8.	*Розробка програмного забезпечення для прогнозування*
9.	*Розробка програмного забезпечення для створення фінального звіту*
10.	*Аналіз даних контрольного прикладу та тестування програми*
11.	*Оформлення пояснювальної записки та додатків*
12.	*Оформлення графічної частини та презентаційних матеріалів комп'ютерного захисту*
13.	*Перед захист кваліфікаційної роботи*
14.	*Представлення на рецензування*
15.	*Представлення кваліфікаційної роботи в ДЕК*

5. Дата видачі
завдання_____

Студент _____ О. О. Безуглий
підпис ініціали, прізвище

Керівник роботи _____ Є. С. Меняйлов
підпис ініціали, прізвище

ЗМІСТ

ВСТУП

В умовах сучасного інформаційного суспільства значну роль відіграє інструментарій аналізу даних в галузі цифрового маркетингу для підприємницької діяльності будь якого масштабу, тому задача розробці програмного забезпечення аналізу, прогнозування та візуалізації даних **є актуальною**. Проблема структурованого зберігання великих обсягів даних, моніторинг та оперативне автоматичне оновлення даних, розрахунок ключових показників ефективності маркетингових заходів потрібує розробки програмного забезпечення, яке дозволяє гнучке формування прогнозних оцінок та візуалізації результатів маркетинг-аналізу для фахівців або осіб, що приймають рішення щодо стратегії підприємницької діяльності.

Цифровий маркетинг має наступні відмінні риси: масштаб операційної діяльності не є залежним від розміру і місцезнаходження; інформація є відновлювальним ресурсом, який може бути використаний багато разів; внаслідок підвищеної інформованості споживачів їх поведінка стає активною. Прогнозна аналітика допомагає робити прогнози на майбутнє, аналізувати дані про відвідувачів, оцінювати тенденції та розробляти нові стратегії. На основі цього аналізу можна спрогнозувати очікуваний прибуток, оцінити ефективність рекламних заходів, що особливо важливо для зростаючих компаній. Але для реалізації алгоритмів прогнозування потрібен достатній обсяг даних, тому алгоритми прогнозування можуть бути обмежені в використанні у компаніях, які тільки починають свою діяльність, через брак даних. В роботі використовується дані про рекламу на онлайн сервіси, які отримані в соціальній мережі Facebook за достатньо тривалий час з щоденним їх оновленням, дані про підписки на онлайн сервіси в їх динаміці.

Таким чином **актуальність робо**ти полягає в унікальності та затребуваності розробленої в дипломної роботі прототипу інтегрованої програмної системи аналітики для Інтернет-маркетингу.

Предметом дослідження є алгоритми прогнозування та візуалізації динамічних даних для Інтернет маркетингу з застосуванням хмарних технологій обробки даних.

Метою роботи Метою роботи є розробка інтегрованої програмної системи аналітики для Інтернет-маркетингу, побудова алгоритму прогнозування показників очікуваного прибутку та його програмна реалізація, інтеграція розрахунків в систему за допомогою конвеєра даних, дослідження BI інструментарію для візуалізації отриманих результатів та налаштування відображення результатів прогнозування та визначених метрик маркетингового аналізу даних для кампанії, яка надає продукт за умовою підписки на визначений термін.

Об'єктом дослідження виступає розробка інтегрованої програмної системи для Інтернет -маркетингу

Задачі дослідження:

– аналіз особливостей методів та показників Інтернет маркетингу для SaaS бізнес моделі;

– дослідження моделей формування прогнозної оцінки доходу кампанії з реалізацією бізнес моделі за підпискою на основі моніторингу даних клієнтів з урахуванням типу продукту або послуги, термінів передплати, місця знаходження клієнта, визначення джерела надходження інформації;

– дослідження та проектування структури та функціональних вимог до розроблювальної інтегрованої програмної системи;

– розроблення алгоритму та програмна реалізація формування прогнозних оцінок коефіцієнтів утримання клієнтів на основі зміщеної бета-геометричної моделі для розрахунку очікуваного доходу за рахунок продовження передплат клієнтами;

– розроблення структури і проведення аналізу моделі даних системи, необхідних для побудові прогнозних оцінок;

– розроблення скрипту для побудови наборів даних з розрахунковими даними прогнозів в загальної задачі конвеєризації даних з використанням хмарних сервісів та оркестратору Apache Airfow;

– дослідження Business Intelligence інструментарію для візуалізації отриманих результатів;

– розробка та налаштування гнучкого відображення для візуалізації отриманих результатів за допомогою системи BI Tableau .

В результаті проведеної роботи в межах дипломного проекту розроблено прототип програмної системи, яка на основі підготовлених даних обчислює прогнозні оцінки коефіцієнтів утримання клієнтів та прогнозні оцінки доходу для кампанії, котра надає можливість користуватися певними сервісними функціями за умови оформлення підписки на визначений термін. Розроблена програма для формування структури даних з результатами розрахунків, яка інтегрована за допомогою оркестратору Apache Airflow в систему. Опановано методи гнучкої візуалізації метрик маркетингового аналізу, прогнозних оцінок в аспектах, визначених фахівцем з маркетингу в системі BI Tableau. Отримані результати можуть використовуватися фахівцем для оцінки ефективності проведених рекламних заходів в соціальний мережі Facebook, формування подальшої стратегії розвитку бізнесу.

Розроблений прототип інтегрованої системи для Інтернет маркетингу забезпечує:

- безперебійну обробку даних, обчислення прогнозних показників очікуваного доходу, збереження інформації, яка постійно оновлюється в системі;
- динамічне відображення результатів в наочному графічному, географічному та кількісному поданні.

Областю застосування можуть виступати інформаційні системи, які надають SaaS або DaaS послуги за умови оформлення придбання ресурсу або підписки на визначений термін.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Інтернет-маркетинг сьогодні є одним з найперспективніших напрямів розвитку маркетингу в Україні, який відрізняється особливістю швидкого розповсюдження інформації, не вимагає великих матеріальних ресурсів та витрат і має можливість широкого охоплення активної цільової аудиторії за мінімально короткий термін[1] .

Традиційні елементи маркетингу (продукт, розповсюдження, просування, маркетингові досліджені) за допомогою інформаційних технології обробки даних дозволяють застосування засобів Інтернету у віддаленому інтерактивному режимі (дивись рисунок 1.1) [2] .

Рисунок 1.1. Основні цілі запровадження Інтернет маркетингу

Застосування інструментів Інтернет-маркетингу є новітнім підходом до ведення бізнесу з мінімальними витратами, в тому числі з можливістю: використовування результатів різноманітних маркетингових досліджень, контролювання та планування витрат на рекламу, гнучкого реагування на реакцію клієнтів, проведення моніторингу зворотного зв’язку на основі обробки великого обсягу даних [3,4].

В роботі досліджуються методи формування прогнозної оцінки в галузі Інтернет-маркетингу Вирішується задача розробки інтегрованої програмної системи аналітики для Інтернет-маркетингу з метою побудові алгоритмів прогнозування показників ефективності реклами та очікуваного прибутку, з можливістю їх візуалізації на основі моніторингу даних.

Теоретичними основами використання інформаційних технологій у маркетинговій діяльності є створення альтернативних шляхів розв'язання управлінських проблем в умовах, коли показники діяльності компанії є непередбачуваними в результаті складних соціально-економічних взаємодій. Систематизація основних передумов використання інформаційних технологій у маркетинговій діяльності підприємства представлена на рисунку 1.2.

Рисунок 1.2. – Передумови використання ІТ у маркетинговій діяльності

У ході дослідження ринку вивчаються прогнози його розвитку, експертні оцінки ринкових тенденцій, визначаються ключові фактори успіху на досліджуваному ринку. Метою такого дослідження є визначення найбільш ефективних способів проведення конкурентної політики на існуючому ринку, а також оцінка можливості виходу на нові ринки, сегментація ринку [5]. Дослідження продажів спрямоване на визначення найбільш ефективних способів, способів і засобів просування товару. Видобуток знань можна визначити як пошук і вивчення маркетингової інформації. Для вирішення цих

дослідницьких завдань використовуються наступні підходи: автоматичний пошук і аналіз даних на веб-сайтах, інтелектуальний аналіз даних при виявленні і вивченні інформації, пов'язаної з інтересами користувачів до продуктів і послуг.

Одним з нових методів роботи з інформацією, який набуває все більшого поширення, є маркетинг на основі великого обсягу даних, який став особливо популярним у зв'язку з переходом від масового маркетингу до цільового. В умовах жорсткої конкуренції об'єктом пильної уваги є кожен окремий споживач, з яким необхідно налагоджувати постійний зворотний зв'язок. Тому сучасні бази даних - це не просто адресний список клієнтів, як це було раніше, а повна інформація про поведінку споживачів за відносно тривалий період. Відповідно, обсяг даних, що надходять в інформаційну систему, значно зростає. Вміст бази оновлюється з кожною наступним придбанням продукту, проведенням нового рекламного заходу, наявністю та фіксацією кликів у відповідній рекламі, тоді компанія має можливість відстежувати поведінку кожного окремого клієнта в часі. Для обробки маркетингової інформації сі Data Mining. Провідними виробниками інтелектуальних систем бізнес-аналітики є IBM, Oracle, SAP AG і Microsoft .

Потрібно відмітити, що однієї з проблем є оцінка ефективності маркетингової діяльності в соціальних мережах. При зростанні масштабів використання соціальних мереж (на території США за рік Facebook залучив 150 млн., LinkedIn — 41 млн., Twitter — 40 млн., Google+ — 29 млн., Pinterest і Instagram — по 25 млн. відвідувачів; час відвідання соціальних мереж вже перевищив час відвідання сайтів) зрозуміло, що скупчення потенційної цільової аудиторії на певних мережевих майданчиках дозволяє проводити там маркетингові дослідження та комунікаційні заходи. Проте ефективність рекламних заходів важко оцінити за рахунок невизначеності кількісних метрик, що робить неможливим робити прогнозні оцінки. [6].

Сховища даних дозволяють інтегрувати та консолідувати дані з різних джерел. Дані можуть надходити за допомогою мобільних технології, хмарних

технології, соціальних медіа. Сховище даних – це база даних для конкретної предметної області, спеціально розроблена та призначена для підтримки прийняття організаційних рішень. Загальна архітектура систем зберігання та обробки даних надана на рисунку 1.2.

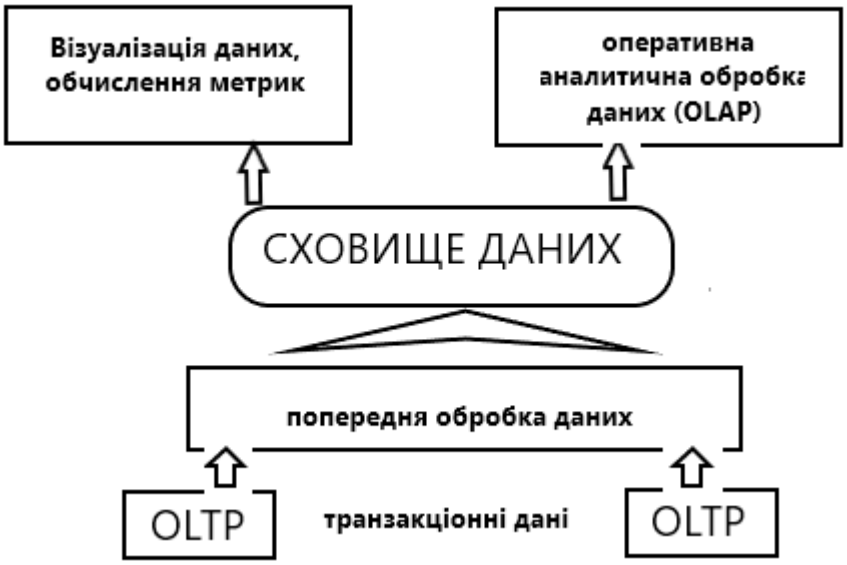


Рисунок 1.3. – Архітектура систем зберігання та обробки даних інформаційної системи

Технології OLAP використовуються для аналізу даних і розробки сценаріїв для моделювання даних, деталізації деталей і узагальнень, фільтрації, сортування та перевпорядкування даних під час аналізу. OLAP - це технологія обробки даних, яка полягає в підготовці зведеної (агрегованої) інформації на основі великих масивів даних, структурованих за багатовимірним принципом.

Технології Data Mining дозволяють використовувати інтелектуальні інструменти бізнес-аналітики в цифровому маркетингу для вирішення різних завдань: багатовимірного аналізу обсягів продажів, витрат на маркетинг та інших змінних за допомогою OLAP; прогнозування ефективності та доходу продажів за допомогою методів регресійного аналізу; використання методів

оптимізації для вирішення завдань оптимізації асортименту, оцінки ефективності та оптимізації маркетингових кампаній; оптимізаційне управління ціновою політикою; завдання класифікації споживачів; виявлення асоціативних правил у споживчому попиті та їх використання для збільшення продажів; сегментація ринку за допомогою методів кластерного аналізу та інші.

1.1 Аналіз існуючих систем та методів аналізу маркетингових даних

Для аналізу маркетингової інформації використовується комплекс економіко-математичних методів, серед яких регресійний, дисперсійний, факторний, дискримінантний, кластерний аналіз [7].

Дисперсійний аналіз використовується для пошуку залежності в експериментальних даних на основі дослідження значущості різниць в середніх значеннях та може використовуватися для оцінки ступені впливу вибору каналу збиту на обсяг збуту.

Факторний аналіз є інструмент для вивчення складних даних, що дозволяє виявити основну структуру інформації та виділити важливі фактори. Цей метод дає можливість більш глибоко вивчити взаємозв'язки між змінними і зрозуміти, як вони впливають на кінцевий результат, наприклад, для виявлення факторів, які впливають на збільшення продажу продукту.

Методи багатовимірного угрупування дозволяє проаналізувати та візуалізувати дані за визначеними шкалами ознак об’єктів, що досліджуються. Якщо групи формуються на основі близькості об’єктів за трьома і більше ознаками, угруповання називається багатовимірним. Цей метод застосовують для розрахунку інтегральних показників на основі багатовимірної середньої, для виконання кластерного аналізу. Класифікація об’єктів проводиться не послідовно за окремими ознаками, а одночасно за кількома ознаками. Ці ознаки утворюють «простір ознак». Як і комбінаційне угруповання, метод багатовимірного угруповання дає змогу систематизувати дані, тобто виділяти

однорідні групи або класифікувати явища, об’єкти, процеси, виконувати рейтингування, на основі якого приймають управлінські рішення [5]. При дискримінантому аналізі створюється прогностична модель членства в групі. Ця модель будує дискримінанту функцію (або, коли груп більше двох, набір дискримінантних функцій) у вигляді лінійної комбінації змінних-предикторів, що забезпечує найкраще розділення груп. Ці функції ґрунтуються на наборі випадків, для яких відома їх групова приналежність, а потім можуть бути застосовані до нових спостережень з відомими значеннями змінних-предикторів, але невідомою груповою приналежністю.

Прогнозування в маркетингу ґрунтується на аналізі даних і використанні статистичних моделей для прогнозування майбутніх тенденцій і подій. Це дозволяє компаніям приймати обґрунтовані рішення щодо ресурсів, розробки продукту та маркетингових стратегій. Наприклад, прогнозування може бути використано для прогнозування попиту на продукцію в певний період, що допомагає у плануванні виробництва та запасів. Для прогнозування використовуються, наприклад, методи регресійного аналізу, методи аналізу динамічних рядів.

В дипломної роботі для побудові прогнозної оцінки коефіцентів утримання клієнтів для обчислення очікуваного доходу досліджується зміщена бета-геометрична модель [8]

1.2 Показники Інтернет- маркетингу

Однією з основних задач кожної системи маркетингового аналізу є автоматизація моніторингу й прогнозуванню показників. Цифровий маркетинг – це виклик сучасності, в той же час багато керівників бізнесу не надають пильної уваги до контролю цього процесу. За дослідженням Boston Consulting Group (BCG), в багатьох , що надають послуги у сфері електронної комерції , не використовують сучасні методи та алгоритми обробки даних та на цієї основі вироблення стратегії розвитку. [9]. Розглянемо основні типи надання

послуг у сфері електронної комерції за підпискою, що досліджується в дипломі.

Data-as-a-Service (DaaS) — це метод керування даними, який використовує хмарні обчислення для надання інформації на вимогу. Слідуючи моделі програмного забезпечення як послуги, де центральний хост доставляє програми кінцевим користувачам на основі періодичної підписки, DaaS організовує інформацію з низки джерел у систему зручних наборів даних, доступних через API. Основна мета DaaS — спростити доступ до даних, що, у свою чергу, сприяє прийняттю обґрунтованих рішень на основі даних[10].

Програмне забезпечення як послуга (SaaS) – це хмарна модель надання програмного забезпечення, яка доставляє програми кінцевим користувачам в Інтернет-браузері. Постачальники SaaS розміщують сервіси та програми, які доступні клієнтам на вимогу. При роботі з моделлю SaaS не потрібно турбуватися про підтримку сервісу або керування базовою інфраструктурою і повністю сконцентруватися на використанні програмного забезпечення. Ще одним типовим аспектом моделі SaaS є ціна. Оплата здійснюється за підпискою або в міру використання, і не потрібно набувати всіх функцій відразу в одному великому пакеті. Найпоширенішим прикладом програми SaaS є інтернет-пошта, користуючись якою ви можете надсилати та отримувати повідомлення, не переймаючись керуванням додаванням функцій у продукт або обслуговування серверів та операційних систем, на базі яких працює електронна пошта. [11]

Загальна схема воронки залучення клієнтів до послуг наведено на рисунку 1.3, в роботі розглядається етап утримання клієнтів (Retention) та досліджуються відповідні показники маркетингового аналізу.

Одним з показників оцінки ефективності витрат на рекламу є показник вартості за дію - Cost Per Action(CPA) [12]. Причому дією вважатися може або перехід на сайт, або підписка, або заповнення форми тощо. Значення CPA-метрик порівнюють з доходом від кожного продажу

Рисунок 1.4 – Воронка продаж

Для розрахунку вартості за дію потрібно взяти суму рекламного бюджету і розділити її на кількість конверсії. Чим нижче значення метрики, тим вигіднішою є рекламна кампанія:

$$CPA = \frac{\text{Витрати на рекламу}}{\text{Кількість дій, виконаних користувачем}}$$

Показник ROI для бізнесу (Return on Investment, повернення інвестицій) - це коефіцієнт рентабельності інвестицій тобто окупність вкладень. Показник демонструє, наскільки вигідним чи невигідним є проект чи продукт.

$$ROI = \frac{\text{Дохід з проекту} - \text{Витрати на проект}}{\text{Витрати на проект}} \times 100\%$$

Якщо значення показника перевищує 100%, то це свідчить про ефективність інвестицій у рекламну кампанію та результативність бізнес стратегії кампанії.

Показник прибутковості, що оцінює прибуток компанії з кожного клієнта Average Revenue Per Account (APRA) –середня виручка на клієнта. У дипломній роботі розраховується за період один день, тиждень, місяць та має назву Average Revenue per Subscriber (ARPS) . Ця метрика є обов'язковою для компаній, які працюють на підписній бізнес-моделі. Основні сфери застосування ARPS:

- порівняння результатів компанії з результатами компаній-конкурентів;
- сегментація покупців у межах послуги (продукту) чи клієнтів;
- розрахунок поточного доходу та його прогнозування.

На цей показник можуть вплинути такі фактори:

- передплата клієнтів на різні доповнення, що оплачуються окремо;
- клієнти, які перейшли на більш дешевий план оплати передплати;
- клієнти, які перестали користуватися послугами компанії протягом розрахункового періоду[13].

Розрахунок ARPS відбувається за формулою:

$$\text{ARPS} = \frac{\text{Загальний дохід підприємства}}{\text{Кількість користувачів, які купили повний SaaS пакет продукту}} \times 100\%$$

Коефіцієнти відтоку (Churn Rate) — це відсоток клієнтів, які відмовляються від послуг чи продукту за певний період. Відтік - це природний процес, при якому бренд втрачає актуальність для конкретного клієнта та клієнти не продовжили співпрацю з кампанією, не зробили повторну передплату для отримання продукту кампанії. Цей показник особливо необхідно відстежувати для бізнесу, який залежіть від постійних клієнтів. Наприклад, коли послуги надаються в форматі підписки. Якщо цей значення показника невелике, то відповідно, клієнти продовжують користуватися послугами або продуктами кампанії.

В дипломі цей показник (Churn Rate або Cancel Rate) розраховується в динамці за різні періоди: на добу, тиждень та місяць[14].

$$\text{Churn Rate} = \frac{\text{Кількість клієнтів, які за період пішли}}{\text{Кількість клієнтів на початку періоду}} \text{ x } 100\%$$

Коефіцієнт відшкодування — це фінансовий показник і показник гарантії якості, який стосується загальної кількості транзакцій або продуктів, які були продані та згодом відшкодовані. Коефіцієнт відшкодування платежів відображає суму, яка повертається продавцем покупцю внаслідок анулюції останнім платежу за зроблену покупку. Таким чином, даний індикатор показує частку клієнтів, що відмовилися від використання купленого продукту[15]:

$$\text{Refund Rate} = \frac{\text{Сума (або кількість) повернених платежів}}{\text{Загальна сума (або кількість) клієнтських платежів}} \text{ x100\%}$$

Цінність клієнта для бізнесу у фінансовому вираженні, а саме загальна сума доходу, яку принесе клієнт компанії за весь час їх взаємодії, має назву - показника життєвій цінності клієнта (Lifetime Value). Визначення цього показника дозволить оцінити успішність стратегії бізнесу щодо просування товарів та послуг, дізнатися про переваги клієнтів, обирати найвигідніші канали, сформувати прогнозні оцінки доходу, визначити сегменти клієнтської бази.

Одним з головних показників, що розраховується в дипломі це коефіцієнт рівня утримання клієнтів «Customer Retention Rate» ((CRR)-, що відображає який відсоток клієнтів залишається з компанією по закінченню

заздалегідь визначеного періоду (в роботі використовується термін Rebill-коефіцієнти)[16].

$$CRR = \frac{\text{Кількість клієнтів в кінці періоду — нові клієнти}}{\text{Кількість клієнтів на початку періоду}} \text{ x100\%}$$

1.3 Аналіз BI систем для візуалізації даних

Для візуалізації та аналізу даних використовуються BI-системи, серед найбільш популярних платформ систем BI можна зазначити: Qlik, Microsoft, Sisense, Tableau, Zoomdata, Information Builders.

Сучасні BI-системи (Business Intelligence)– це інструмент обробки і аналізу будь яких даних, серед функцій таких систем можна зазначити наступні: робота з різними джерелами даних та з обробкою взаємозв'язків між даними; побудова звітів з різною структурою як вручну, так і автоматично за розкладом; інтерактивна робота з даними; можливість представлення реляційних даних як багатовимірних; автоматична розсилка сформованих звітів у різних форматах.

Систему бізнес-аналітики використовують коли потрібно: аналізувати фінансові показники, оптимізувати операції за рахунок аналізу маркетинг даних; покращити продажі, аналізуючи дані про клієнтів, продажі, рекламу та конкурентів; прийняти рішення на основі даних про подальший розвиток бізнесу.

В дипломної роботі використовується система Tableau — це програмне забезпечення для візуалізації даних та аналітичної звітності, яке дозволяє створювати інтерактивні звіти, діаграми, графіки та інші типи візуалізацій для аналізу великих обсягів даних.

Tableau — це провідний інструмент бізнес-аналітики (BI) і візуалізації даних, розроблений, щоб зробити аналіз даних доступним та інтуїтивно

зрозумілим для користувачів різного рівня кваліфікації. Він дає змогу перетворювати необроблені дані в інтерактивні інформаційні панелі, якими можна ділитися, надаючи інформацію, яка спонукає до прийняття обґрунтованих рішень.[9]

Приклад інтерфейсу Tableau наведено на рисунку 1.5.

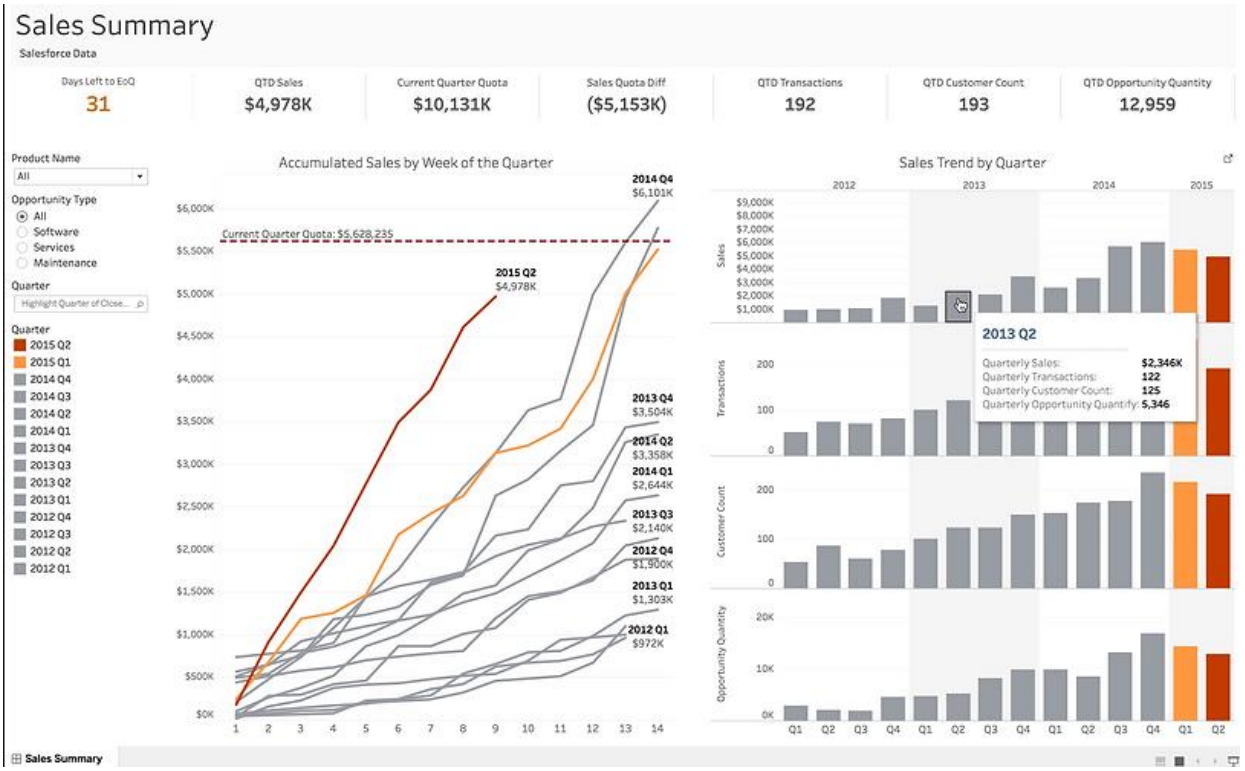


Рисунок 1.5. – Приклад інтерфейсу системи BI Tableau

Tableau швидко зарекомендував себе на ринку бізнес-аналітики, професіонали всіх відділів можуть розуміти дані та інтерпретувати отримані аналізи даних. Він орієнтований насамперед на експертів з обробки маркетингових даних.[17,18]

На відміну від традиційних BI-інструментів, які вимагають глибоких технічних знань, Tableau надає перевагу зручності для користувача, дозволяючи як технічним, так і нетехнічним користувачам створювати складні візуалізації та аналізи з легкістю. Він підтримує широкий спектр джерел даних, від електронних таблиць і баз даних до хмарних служб, забезпечуючи гнучкість і підключення.

В межах дипломної роботи дані для аналізу отримуються від соціальній мережі Facebook та з backend-програми з сервісними послугами, на яку підписаний клієнт. Аналіз активності споживачів надані у вигляді неструктурованих Інтернет-даних, які зберігаються в сховищах даних за допомогою хмарних технологій.

1.4 Постановка задачі

Метою роботи є розробка інтегрованої програмної системи аналітики для Інтернет-маркетингу для побудові алгоритму прогнозування показників очікуваного прибутку, з можливістю візуалізації визначених метрик маркетингового аналізу даних за допомогою конвеєра даних для кампанії, яка надає можливість користуватися певними сервісними онлайн функціями за умови оформлення передплати на визначений термін.

Отже потрібно виконати наступні задачі:

– розробити та уточнити структуру та функціональні вимоги до програмної системи;

– реалізувати зміщену бета-геометричну модель формування прогнозної оцінки очікуваного доходу за рахунок продовження передплат клієнтами;

– оцінити точність отриманих прогнозних оцінок;

– розробити структуру і проаналізувати моделі даних системи, необхідних для побудові прогнозних оцінок;

– розробити скрипт для побудови таблиці даних з розрахунковими даними прогнозів в загальної задаче конвеєризації даних;

– розробити програмне забезпечення та провести реалізацію проектованої системи;

– розрахувати кількісні метрики показників маркетингового аналізу;

– розробити засоби для візуалізації отриманих результатів за допомогою системи BI Tableau .

Система повинна забезпечувати безперебійну обробку, збереження вхідної інформації, яка постійно оновлюється в системі.

Вхідною інформацією є дані про витрати на рекламу, отримані з мережі Facebook, дані про клієнтів; дані про існуючі передплати на онлайн-ресурс; терміни передплат та закінчення передплат; вартісні характеристики передплат; географічні ознаки розташування клієнтів; ймовірнісні коефіцієнти не продовження передплат клієнтів на послуги.

Вихідною інформацією системи є обчислені прогнозні коефіцієнти продовження передплат клієнтами на певний термін, поточний доход, прогнозні оцінки доходу, обчисленні показники маркетингової аналітики та візуалізація даних засобами системи BI Tableau.

В дипломної роботі потрібно обчислити наступні маркетингові показники:

– кількість відмовлень від передплат за один день/за тиждень/за місяць (Cancel Rate(Churn Rate) 1/7/30d);

– фактичний дохід на передплатника (Actual Revenue per Subscriber - ARPS), показник ARPS корисний для порівняння груп або когорт облікових записів за місяцями з метою для виявлення тенденцій у розширенні та скороченні облікового запису, оцінки планів ціноутворення і розуміння того, як розвивається бізнес;

– показник очікуваного повернення інвестицій (Return of Investments - ROI) - це співвідношення грошової суми, витраченої на маркетингові кампанії, і доходу, що вони забезпечили;

– CPA («Cost Per Action», вартість дії) — показник, який допомагає компанії розрахувати, скільки коштувало залучення одного користувача, який завершив цільову дію на сайті.

– CRR («Customer Retention Rate») -показник утримання клієнтів.

2 МАТЕМАТИЧНА МОДЕЛЬ ФОРМУВАННЯ ПРОГНОЗНИХ ОЦІНОК

В основі будь-якої бізнес-моделі, орієнтованої на контракт або підписку, лежить поняття коефіцієнта утримання клієнтів. В дипломної роботі використовується зміщена бета-геометрична модель для прогнозування коефіцієнтів утримання клієнтів. Одним з управлінських завдань є проектування показників утримання клієнтів для певної групи клієнтів на основі ряду минулих показників на майбутнє, щоб зробити більш точні прогнози щодо терміну перебування клієнтів на сайті, їхньої життєвої цінності і подальшого оцінювання доходу кампанії. Визначальною характеристикою контрактного або передплатного бізнесу є те, що спостерігається відхід клієнта. Наприклад, клієнт повинен зв'язатися з фірмою, щоб скасувати контракт на мобільний телефон; аналогічно, місцева театральна компанія може помітити, що відвідувач не поновив свій річний абонемент. Отже, доцільно дослідити такі показники, як коефіцієнт утримання та відтоку: коефіцієнт утримання для періоду t визначається як частка клієнтів, активних на кінець періоду t-1, які все ще активні на кінець періоду $t(r_t)$, в той час як коефіцієнт відтоку для даного періоду визначається як частка клієнтів, активних на кінець періоду t-1, які припинили свою діяльність у періоді t.

В таблиці 2.1 наведений приклад розрахунку коефіцієнтів утримання клієнтів на кожен наступний період підписки в кампанії, у якої було на початок першого року 1000 передплатників [19] . В дипломної роботі потрібно розрахувати прогнозні ймовірнісні оцінки коефіцієнту утримання клієнтів та відповідні значення очікуваного доходу кампанії. Група клієнтів, які об’єднані датою залучення до підписки має визначення когорти. Коефіцієнт утримання відбиває відсоток клієнтів, які виживають між наступними періодами.

Таблиця 2.1-Приклад розрахунку фактичних значень показників

		S(t)	r(t)
0	1000	1	
1	629	0,629	0,629
2	473	0,473	0,752
3	384	0,384	0,812
4	326	0,326	0,849
5	285	0,285	0,874
6	254	0,254	0,891
7	230	0,23	0,906
8	211	0,211	0,917

Коефіцієнт відтоку доповнює коефіцієнт утримання (тобто 1 мінус утримання). Коефіцієнти виживання відбивають відсоток клієнтів, котрі вижили початку періоду. Функція виживання S(t) — це ймовірність того, що випадково обраний член початкової когорти клієнтів вижив, тобто залишиться та продовжить підписку на наступний термін після часу t. Відповідно для даних таблиці 2.1:

$$S(1)=0,629,\ S(2)=0,473,\ S(3)=0,384,\ldots \qquad (2.1)$$

Коефіцієнти утримання клієнтів можна обчислити безпосередньо з даних наданих у вигляді кількості підписаних клієнтів з когорти за допомогою функції виживання за такою формулою:

$$r_t = \frac{S(t)}{S(t-1)},\ \ t = 1,2,3\ \ldots \qquad (2.2)$$

де функція S(t) — функція виживання . Відповідно до періодів з прикладу коефіцієнти утримання будуть мати значення: r(1)=0,631, r(2)=0,742, r(3)=0,816,…

Аналогічно, маючи знання про показники утримання, ми можемо обчислити функцію виживання за допомогою такої прямої рекурсії:

$$S(t) = \begin{cases} 1, & \text{якщо } t = 0 \\ r(t) \times S(t-1), & \text{якщо } t = 1,2,3,.. \end{cases} . \quad (2.3)$$

Для того, щоб охопити неоднорідність клієнтів припустимо, що ймовірність кожного індивіда може приймати будь-яке з нескінченної кількості можливих значень від 0 до 1; це досягається шляхом припущення, що мінливість цих ймовірностей між клієнтами враховується безперервним розподілом ймовірностей. У якості такого розподілу ймовірностей (що може коливатися від 0 до 1) звертаються до бета-розподілу тому, що він є і гнучким (тобто, він може вловлювати багато різних шаблонів неоднорідності), і простим у роботі під час виконання різноманітних математичних розрахунків. Для бета розподілу математичне сподівання E[X] та дисперсія D[X] обчислюється за формулами:

$$E[X] = \frac{\propto}{(\propto+\beta)} , \quad (2.4)$$

$$D[X] = \frac{\propto\beta}{(\propto+\beta)^2(\propto+\beta+1)} . \quad (2.5)$$

Поєднання цих двох розподілів ймовірностей надає нам бета-геометричний (BG) розподіл як модель тривалості контракту. За моделлю BG отримаємо наступний вираз для коефіцієнта утримання:

$$r(t \mid \propto, \beta) = \frac{\propto+t-1}{\propto+\beta+t-1} . \quad (2.6)$$

Враховуючи (2.2), функцію виживання S(t), можна обчислити використовуючи пряму рекурсію, наведену в (2.1) отримаємо наступні формули:

$$S(0)=1,$$
$$S(1)=S(0)\times r(1)=\frac{\alpha}{\alpha+\beta},\qquad (2.7)$$
$$S(2)=S(1)\times r(2)=\frac{\alpha}{\alpha+\beta}\times\frac{\alpha+1}{\alpha+\beta+1},\ldots$$

В таблиці 2.2 наведено приклад обчислення модельних значень коефіцієнтів утримання клієнтів та відповідної функції виживання. Параметри моделі зміщеного бета-геометричного розподілу були отримані шляхом оптимізації за допомогою алгоритму Нелдера-Мида (бібліотечна функція scipy.optimize.minimize Minimization of scalar function of one or more variables using the Nelder-Mead algorithm) та для досліджуваного прикладу мають наступні значення: α=1,36; β=0,79.

Таблиця 2.2 – Обчислення значень моделі

		S(t)	r(t)	r(t)m	Sm(t)
0	1000	1			
1	629	0,629	0,629	0,629	0,629
2	473	0,473	0,752	0,748	0,471
3	384	0,384	0,812	0,809	0,381
4	326	0,326	0,849	0,846	0,322
5	285	0,285	0,874	0,871	0,281
6	254	0,254	0,891	0,889	0,250
7	230	0,23	0,906	0,903	0,225
8	211	0,211	0,917	0,913	0,206
9	195	0,195	0,924	0,922	0,190
10	181	0,181	0,928	0,929	0,176
11	170	0,17	0,939	0,935	0,165
12	160	0,16	0,941	0,940	0,155

2.1 Оцінка точності моделі

В дипломної роботі для обчислення точності отриманих прогнозних оцінок об використовувались оцінки загального середнього, середньоквадратична похибка, коефіцієнт детермінації.

Середньоквадратична похибка визначається як квадратний корінь зі всіх квадратів відстані, поділений на загальну кількість точок та використовується для визначення наявності великих помилок або відстаней, які можуть бути викликані, якщо модель переоцінила інтерполяцію (тобто прогнозовані моделі значення, які були значно вищі за фактичне значення) або недооцінила інтерполяцію (т. е. прогнозовані значення менше фактичного прогнозу[12]:

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2} \tag{2.8}$$

Якщо вектор з n прогнозованих значень обчислюється з вибірки n даних, тобто якщо у є вектором спостережуваних значень передбачуваної змінної, а $\hat{y}_t$ є прогнозними значеннями , то середньоквадратична помилка розрахунків передбачення (MSE) в межах цієї вибірки обчислюється як:

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2 \tag{2.9}$$

Коефіцієнт детермінації R^2 характеризує частку варіації залежної змінної, обумовленої мінливістю пояснюючих змінних; чим ближче R^2 до одиниці, тим краще описується залежність між пояснювальною та залежною змінними:

$$R^2 = 1 - \frac{ESS}{TSS} \quad , \tag{2.10}$$

де відповідно: $ESS = \sum_{t=1}^{n}(y_t - \hat{y}_t)^2$ - сума квадратів похибок моделі відносно фактичних даних, тобто сума квадратів залишків; $TSS = \sum_{t=1}^{n}(y_t - \bar{y}_t)^2$ - загальна сума квадратів відхилень від середнього значення; y_t –значення, що прогнозується , $\bar{y}$ – середнє значення, $\hat{y}_t$ – значення по моделі.

Можливі три ситуації із різними інтерпретаціями коефіцієнту детермінації.

1.Якщо $R^2 = 1$ – модель ідеально описала ряд даних. Така ситуація можлива лише в тому випадку, якщо всі розрахункові начення виявились рівними всім фактичним.

2. Якщо $R^2 < 1$ — модель описала відповідний відсоток дисперсії фактичних значень.

3. Якщо $R^2 \notin (0;1)$ — модель має надто високі значення порівняно з фактичними, що можливо при використання нелінійних моделей прогнозу

2.2 Обґрунтування вибору методу sBG

Одним з показників маркетингового аналізу є «пожиттєва цінність клієнта» (Customer Lifetime Value- CLV), що означає чистий прибуток, який приносить клієнт протягом усього періоду свого життєвого циклу в компанії. Щоб оцінити CLV, вам необхідно зрозуміти два простих атрибути клієнта; тривалість життя клієнтів та дохід, який вони приносять на регулярній основі. Чим більше клієнт платить і чим довше він залишається клієнтом, тим вищий його CLV [20] :

$$CLV = \sum_{t=0}^{T} \frac{mr^t}{(1+d)^t} \qquad (2.11)$$

де m- доход за період, r- коефіцієнт утримання за період, d- ставка дисконту за період, T- часовий горизонт

У цієї методології є наступні недоліки:

– нездатність врахувати невизначеність, тому що ця формула обчислює одне число, яке виражає очікувану довічну цінність клієнта (чи середнє значення) , в тому як у SaaS-бізнесі вклад клієнтів у дохід може сильно

відрізнятися, особливо якщо у вас різні плани обслуговування (базовий та преміальний);

– постійний рівень утримання, тобто використання сукупного показника утримання серед ваших клієнтів

– відсутність прийняття до уваги типу передплати: контрактний або позаконтрактний.

У контрактному бізнесі (наприклад SaaS) явно спостерігається відтік клієнтів. Наприклад, більшість клієнтів SaaS працюють за підпискою, і тому клієнту доведеться відмовитись від підписки, щоб відмовитися від неї.

Неконтрактні підприємства (наприклад, інтернет-торгівля, продуктові магазини) не знають, чи була остання покупка клієнта справді останньою чи він зрештою повернеться, щоб зробити ще одну покупку.

В роботі [21] приведений розрахунок різних моделей для динаміки функції виживання клієнтів за даними сьомі років у порівнянні, також проведений аналіз застосування моделей для прогнозу на наступні п’ять місяців. За допомогою регресійного аналізу отримані наступні рівня іта обчислений коефіцієнт детермінації:

$$y = 0{,}773x - 0092 \qquad R^2 = 0.776$$

$$y = 0.022x^2 - 0.249x + 0.930 \qquad R^2 = 0.9697$$

$$\ln(y) = -0.248 - 0{,}190x \qquad R^2 = 0.915$$

На графіку (дивись рисунок 2.1) надані графікі, які відображають збільшення похибки прогнозування.

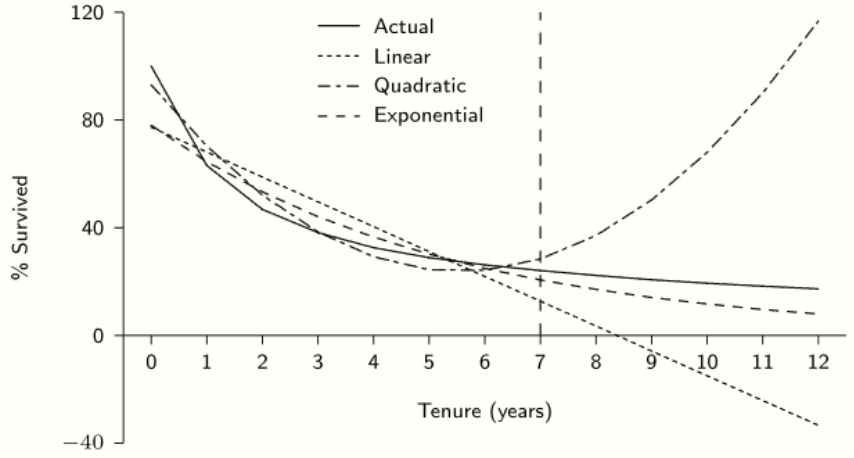


Рисунок 2.1 - Порівняльний графік отриманих моделей прогнозування

Всі три моделі добре узгоджуються між собою до 7-го року включно, а квадратична модель забезпечує особливо хороше узгодження. Але коли ми розглядаємо прогнози, що виходять за межі періоду калібрування моделі, всі три моделі різко погіршуються. Лінійна та експоненціальна моделі недооцінюють виживання на 12-му році життя на 81% та 30% відповідно, тоді як квадратична модель переоцінює виживання на 12-му році життя на 92%. Крім того, моделям бракує логічної послідовності: лінійна модель мала б S(t)<0 після 14 року, а згідно з квадратичною моделлю виживання буде починати зростати з часом, а це неможливо.

За даними контрольного прикладу дипломної роботи був проведений аналіз та побудовано графікі для порівняння з sBG моделлю (дивись рисунок 2.2) [22].

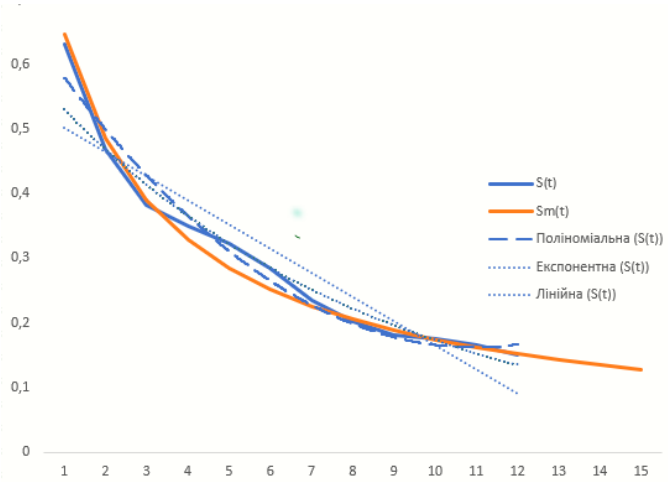


Рисунок 2.2 - Графічне подання результатів контрольного прикладу

На графіку відображені фактичні дані за 12 місяці та прогноз на наступні 3 місяці за методом sBG. За методом регресійного аналізу отримані моделі:

$$y = -0.0375x + 0.5342 \qquad R^2 = 0.867$$
$$y = 0.0042x^2 - 0.0093x + 0.6672 \qquad R^2 = 0.9697$$
$$y = 0.00061e^{-0.124x} \qquad R^2 = 0.9536$$

Всі моделі мають достатньо високий рівень підгонки до фактичних даних, коефіцієнти R^2 знаходяться в межах від 0,867 (лінійна модель) до 0,969

(квадратична модель), але при прогнозуванні похибки збільшуються. Модель sBG за даними контрольного прикладу має $R^2 = 0{,}989$ та більш адаптивна до рівня зниження значень функції виживання/ утримання клієнта.

За даними дипломної роботи для розробки інтегрованої програмної системи був також проведений аналіз застосування моделей прогнозування. На рисунку 2.3 наведено графіки для порівняння моделей за фактичними даними за 6 місяців та отриманими прогнозними оцінками коефіцієнтів утримання клієнтів на наступні три місяці.

За результатами ретроспективного прогнозу фактичних даних, отриманих після обчислення прогнозних значень у порівнянні з розрахованими прогнозними значеннями за моделлю sBG, експоненціальної моделлю можна зробити висновок, що експоненціальна модель є недостатньо адаптивною до темпу зниження коефіцієнтів утримання клієнтів для прогнозування вже на невеликий період часу.

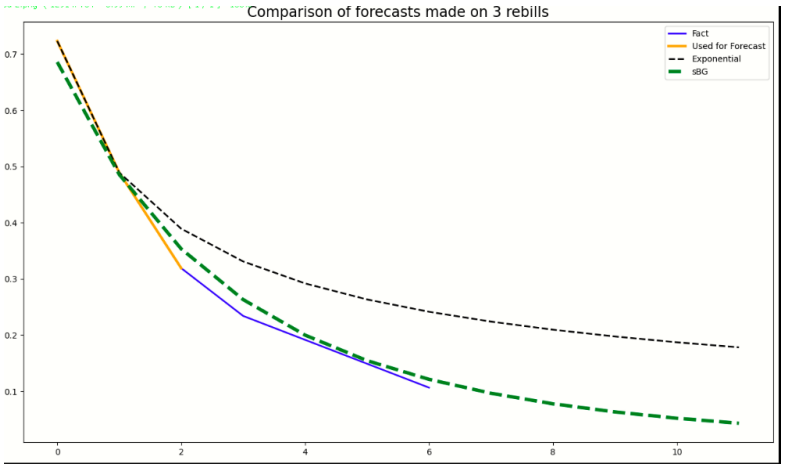


Рисунок 2.3 - Графічне подання прогнозних та фактичних результатів моделювання.

Фрагмент розрахованих значень таблиці показників точності за даними роботи наведено на рисунку 2.4

product_id	country_gro	rebill_number	rebill_rate	weight	created_at	real	forecast	RSE	MRSE
SUB_WEEKLY	1	2	0,831	0	01,02,2022	0,787	0,831	0,002	0,045
SUB_WEEKLY	1	1	1,418	0	01,02,2022	1,445	1,418	0,001	0,032
SUB_WEEKLY	1	0	2,418	0	01,02,2022	2,297	2,418	0,015	0,122
SUB_WEEKLY	0	1	2,089	0	01,02,2022	2,232	2,089	0,020	0,141
SUB_WEEKLY	0	0	3,089	0	01,02,2022	2,917	3,089	0,030	0,173
SUB_WEEKLY	-1	3	0,545	-1	01,02,2022	0,516	0,545	0,001	0,032
SUB_WEEKLY	-1	2	0,906	-1	01,02,2022	0,880	0,906	0,001	0,032
SUB_WEEKLY	-1	1	1,507	-1	01,02,2022	1,427	1,507	0,006	0,077
SUB_WEEKLY	-1	0	2,507	-1	01,02,2022	2,443	2,507	0,004	0,063
SUB_WEEKLY	-2	0	2,507	-1	01,02,2022	2,532	2,507	0,001	0,032
SUB_MONTHLY	1	1	1,222	0	01,02,2022	1,187	1,222	0,001	0,032
SUB_MONTHLY	1	0	2,222	0	01,02,2022	2,320	2,222	0,010	0,100
SUB_MONTHLY	0	3	0,546	0	01,02,2022	0,575	0,546	0,001	0,032
SUB_MONTHLY	0	2	0.873	0	01.02.2022	0.896	0.873	0.001	0.032

Рисунок 2.4 - Фрагмент розрахунку точності прогнозних оцінок

Параметри sBG моделі мають взаємозв'язок з точки зору утримання та відтоку клієнтів. У наведеному нижче рисунку 2.4 класифіковані типи кривих виживання на основі вибіркових ймовірностей втрати клієнтів з використанням бета-розподілу[23].

У кожному квадраті є пара діаграми: лева діаграма являє собою показники відтоку для заданих α і β, а права діаграма являє собою відповідне криву виживання.

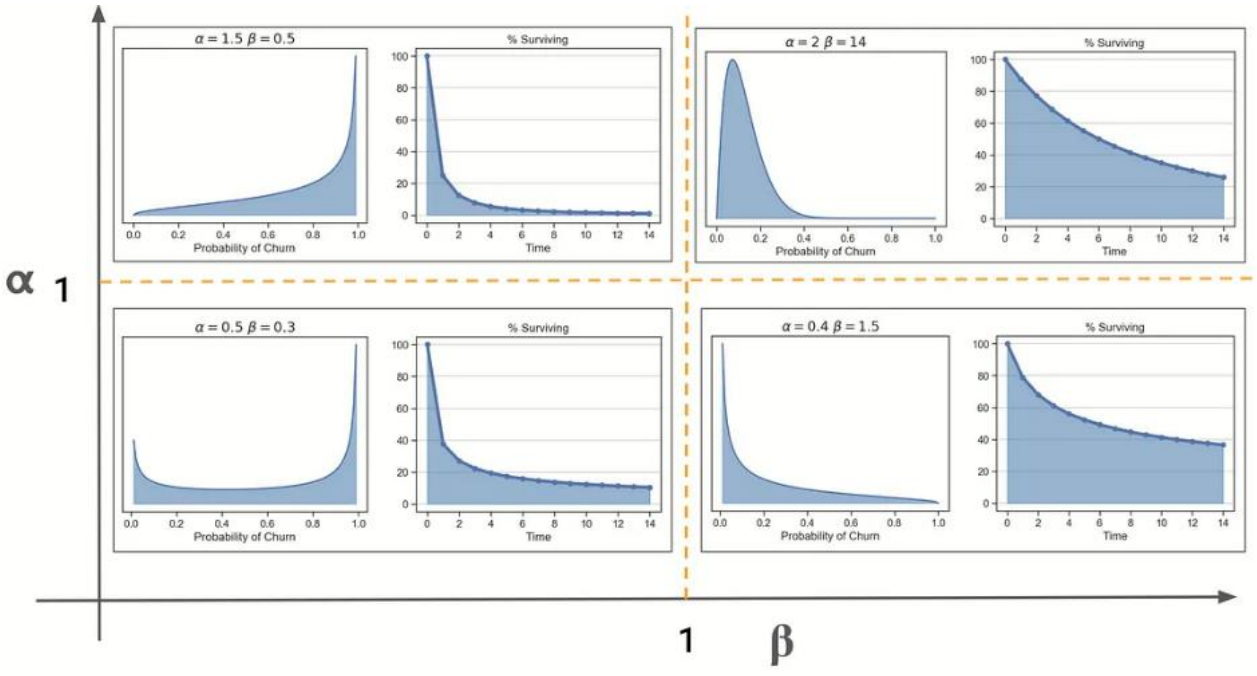


Рисунок 2.5 - Криві виживання з використанням різних розподілів відтоку

Оскільки втрата клієнтів і утримання доповнюють один другого, тоді знання одного дозволяє нам виміряти інше. Бета-розподіл — це гнучкий розподіл, який використовується для моделювання рівня втрати клієнтів з використанням параметрів α і β.

Отже модель sGB надає більше можливостей для моделювання та побудові прогнозних оцінок коефіцієнтів утримання клієнтів для оцінки контрактного та дискретного бізнесу та розрахунку прогнозних оцінок очікуваного доходу.

3 СТРУКТУРНИЙ АНАЛІЗ ПРОГРАМНОЇ СИСТЕМИ

Для розробки інтегрованої програмної системи аналітики Інтернет-маркетингу, а саме для побудові алгоритму прогнозування показників очікуваного доходу, з можливістю візуалізації визначених метрик маркетингового аналізу даних за допомогою конвеєра даних потрібно провести структурний аналіз задачі.

Для моделювання інтегрованої програмної системи (ІПС) Інтернет маркетингу використаємо методологію SADT, стандарт для моделювання функціональних моделей – IDEF0. На контекстній діаграмі (див. рисунок 3.1) представлена задача розробки інтегрованої програмної системи у взаємозв'язку з вхідними, вихідними даними, керуючими впливами, інструментальними засобами її вирішення.

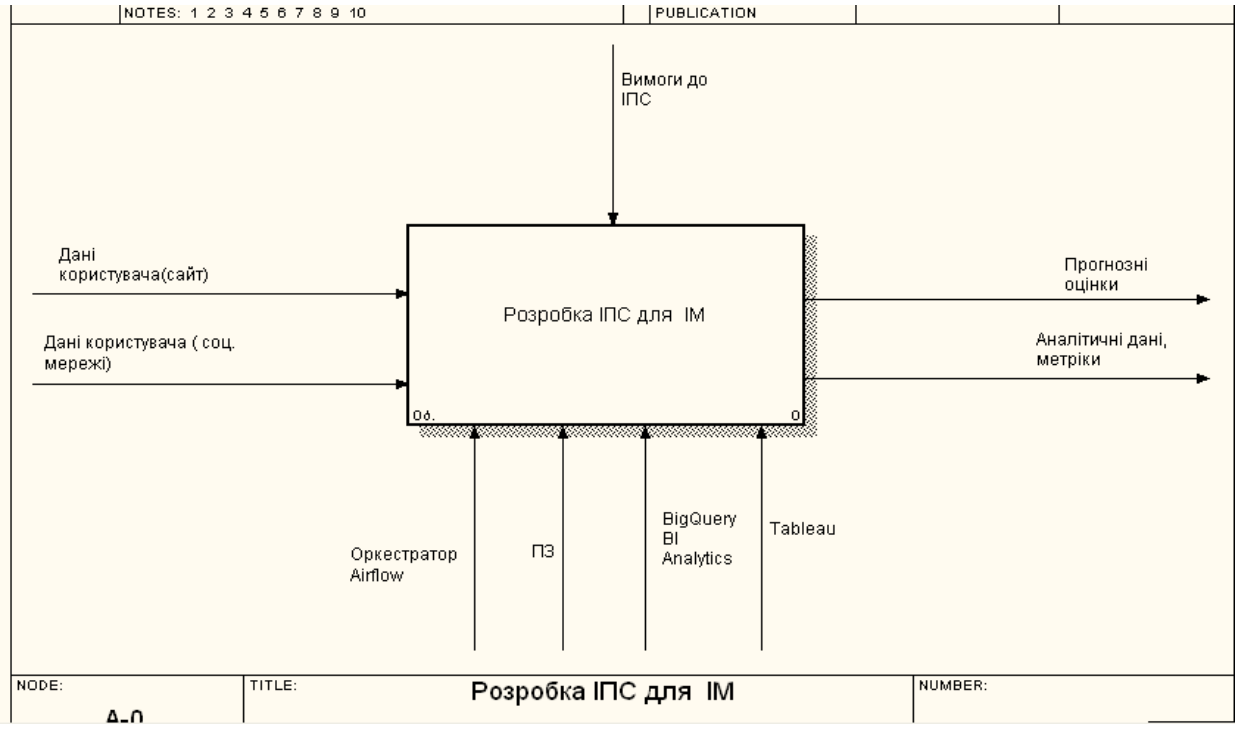


Рисунок 3.1 – Контекстна діаграма

За допомогою функціональної моделі в стандарті IDEF0 надаймо структуру інтегрованої програмної системи з Інтернет маркетингу у вигляді взаємопов'язаних блоків. Стандарт IDEF0 використовується для створення

системи діаграм, які впорядковані в ієрархічному порядку [24]..Мета моделювання – аналіз структури та уточнення функціональних вимог до ІПС, що розроблюється для кампанії, яка надає можливість користуватися певними сервісними онлайн функціями за умови оформлення передплати на визначений термін. Модель надає уявлення з позиції «як-буде» (TO-BE) з точки зору її розробника

Вхідними даними є транзакції з Back-End сайту, на який оформлюється передплата, а саме дані про існуючі передплати на пробний (trial-period) період, дані про передплату на користування на певний термін, дані про терміни передплат щодо кожного користувача; дані з мережі Facebook про затрати на рекламні заходи кампанії; дані про джерело надходження користувача. Вхідні дані автоматично завантажуються у хмарне сховище Google Cloude Storage, оновлюються та обробляються з періодичністю один раз на добу.

Керуючими впливами визначена необхідність дотримуватися законодавчих актів та нормативних документів про обробку персональних даних користувачів та про діяльність кампанії в галузі електронної комерції. Також під час розробки програмної системи необхідно дотримуватись вимог до її функцій, що визначені маркетинговим відділом для проведення моніторингу та аналізу відповідних метрик щодо ефективності інвестицій та розрахунку очікуваного доходу кампанії.

В процесі розробки ІПС для конвеєрної обробки даних використовуються Apache Airflow, Google Cloud Function; для оперативної аналітичної обробки та трансформації даних використовується сховище даних (Data Warehouse-DWH) Google BigQuery; інструментом для візуалізації даних, що дозволяє реалізувати функції BI виступає Tableau. Також у якості інструменту або механізму, за допомогою якого реалізуються функції ІПС позначені алгоритми програмного засобу (ПЗ).

Вихідними даними на контекстній діаграмі позначені:

– обчислені прогнозні оцінки доходу,

– метрики маркетингового аналізу, щодо ефективного витрат на рекламні заходи,

– кількості відказів від передплат за один день/за тиждень/за місяць (Cancel Rate 1/7/30d);

– показники фактичного доходу на передплатника (ARPS);

– показник очікуваного повернення інвестицій (ROI);

– CPA— показник, який допомагає компанії розрахувати, скільки коштувало залучення одного користувача, який завершив цільову дію на сайті.

Розробка ІПС складається з попередньої обробки даних, розрахунку коефіцієнтів подовження передплати, обчислення прогнозних оцінок очікуваного доходу, завантаження даних для візуалізації (дивись рисунок 3.2)

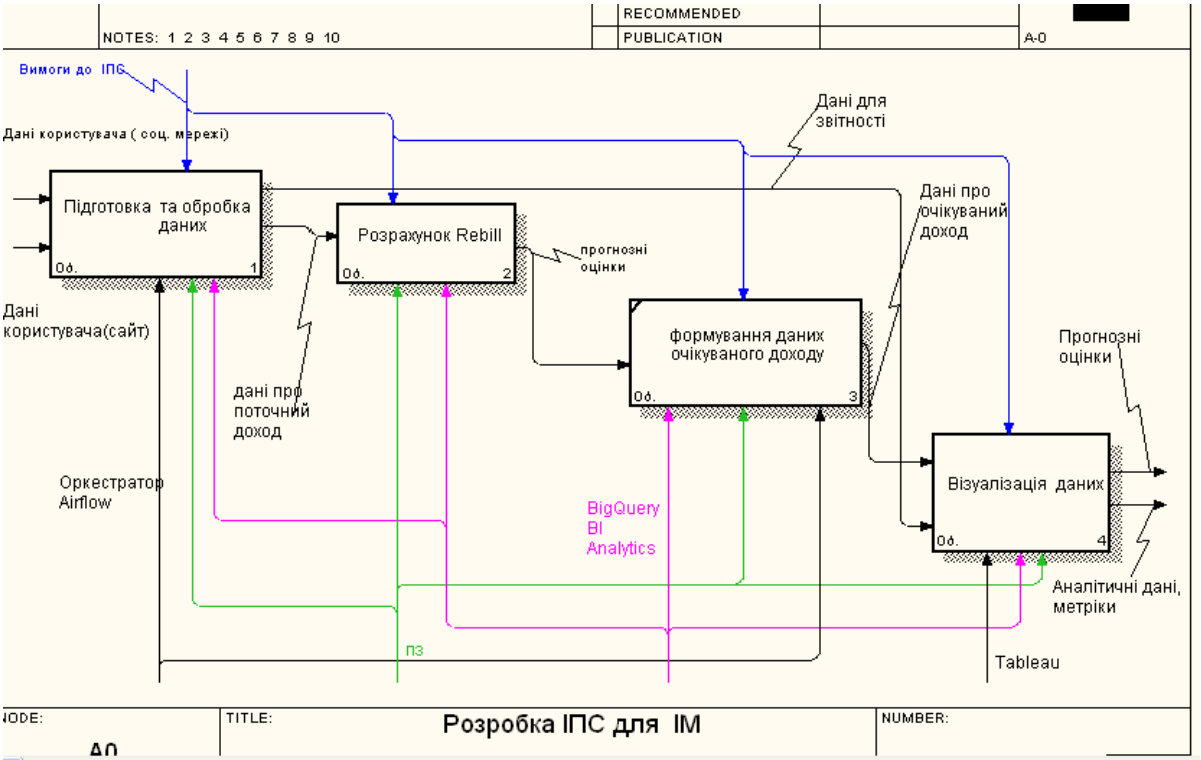


Рисунок 3.2 – Декомпозиція A0- функціональні складові задачі

Функціональні вимоги до попередньої обробки даних представлені на рисунку 3.3. Дані автоматично з джерел надходження зберігаються в Google Cloud Storage. Попередня обробка формує масиви даних за датами. Наступним етапом за допомогою оркестратору Apache

Airflow дані завантажуються раз на добу для зберігання в Big Query. В результаті трансформації та агрегації даних отримуємо інформацію щодо придбань (таблиця замовлення/покупки); статистику про витрати (дані з Facebook); атрибутивні характеристики користувачів, щоб можна з'ясувати звідки прийшов користувач, тобто це органіка або дані с соціальних мереж; дані про передплати користувачів

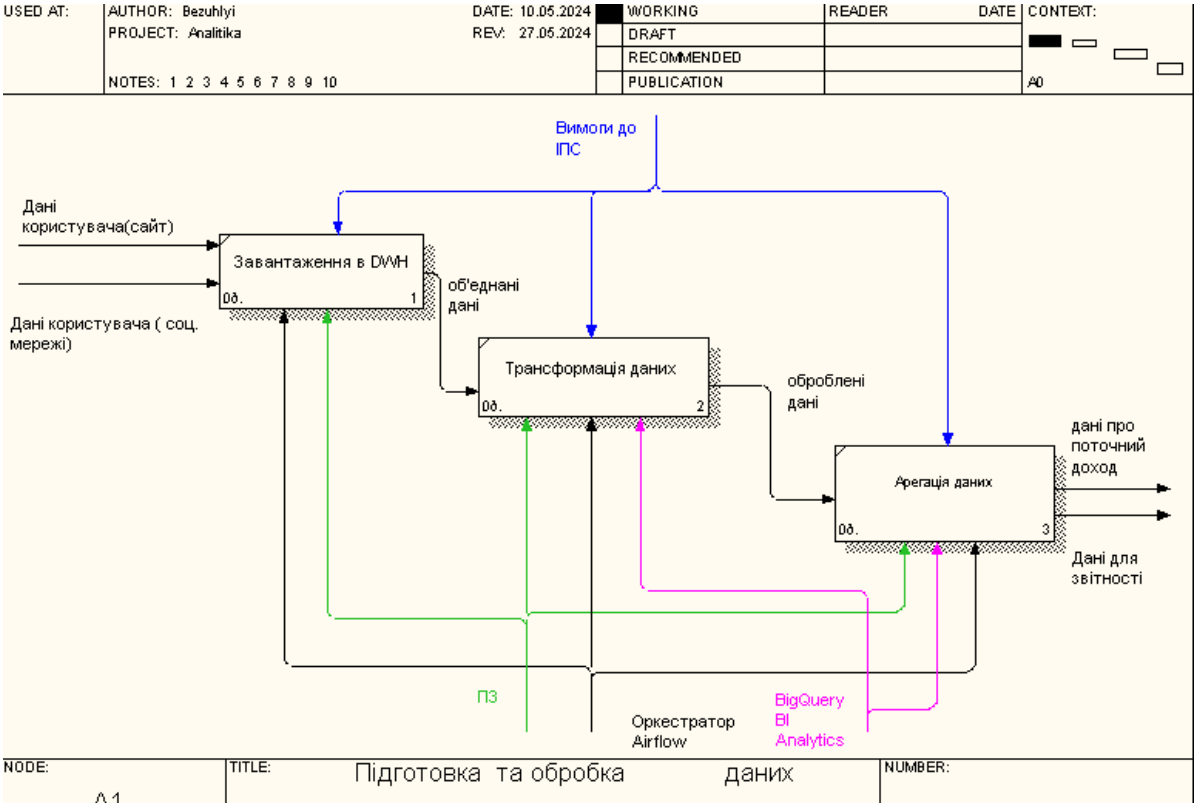


Рисунок 3.3 –Попередня обробка даних

.

Наступний крок полягає в розрахунку прогнозних значень коефіцієнта утримання клієнта Coeffs_Rebill. Розрахунок здійснюється за допомогою зміщеної бета-геометричної моделі прогнозних значень коефіцієнтів утримання клієнтів (Shifted Beta geometric Model -SBG), отримані результати зберігаються в таблицю «Coeffs_Rebill». Основними параметри за якими розраховується коефіцієнти є інформація про продукт і платоспроможність по країнах (Country Group). Основні етапи розрахунку прогнозних значень коефіцієнта утримання клієнта Coeffs_Rebill наведено на рисунку 3.4

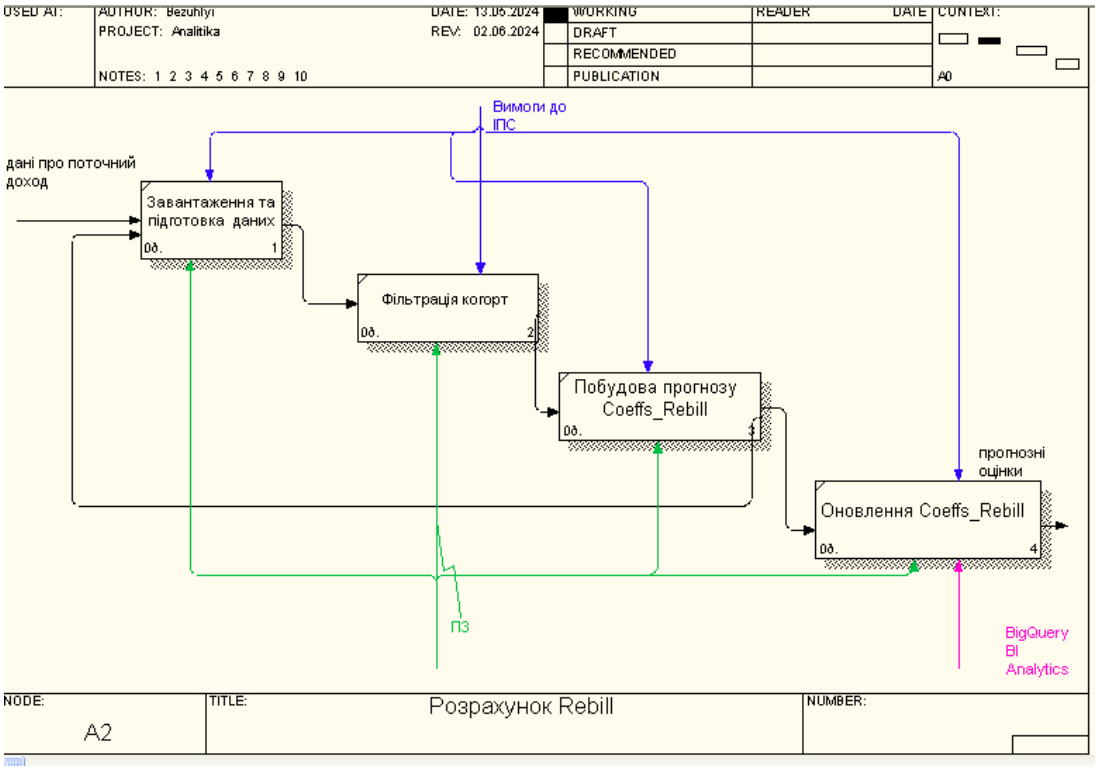


Рисунок 3.4 –Розрахунок прогнозу коефіцієнтів утримання клієнтів

Розрахунок Rebill-коефіцієнтів починається завантаження даних з BigQuery у DataFrame . DataFrame – це двовимірна структура даних мов комп'ютерного програмування. Далі проводиться підготовка даних: дата першого платежу приводиться до формату без урахування часового поясу; розраховується кількість повторних платежів на основі тривалості пробного періоду та періоду повторних платежів; користувачі поділяються на когорти з визначеною тривалістю днів.

Важливим кроком є фільтрування когорт за критерієм наявності можливості користувачам зробити повторні платежі. Формуються групи когорт за часом очікування повторної спроби зробити передплату та за мінімальної кількістю платежів. Розраховується максимальний порядковий номер повторного платежу кожної когорти.

Отримані когорти надходять для формування прогнозних оцінок Rebill-коефіцієнтів, що виконується в циклі за кожен період надходження даних. Цикл побудові прогнозів передбачає угрупування даних та їх підготовку до

розрахунків, визначення ваги даних в межах періоду проєктування, розрахунок прогнозу коефіцієнтів утримання клієнтів для когорт з кількістю повторних платежів не менш ніж два. Розраховані прогнозні оцінки Rebill-коефіцієнтів зберігаються у DataFrame.

На останньому етапі розрахунку виконується вибір та оновлення Rebill-коефіцієнтів, для цього здійснюється вибір коефіцієнтів із найбільшими вагами та оновлення даних: значень коефіцієнтів та накопичених значень повторних платежів. Після додавання інформації про дату створення дані зберігаються в BigQuery:

Для формування даних про очікуваних доход (дивись рисунок 3.2) необхідно враховувати вхідні дані про коефіцієнти, які надають уявлення про те, який відсоток від загального обсягу продажів повертається(Refund-коефіцієнти).

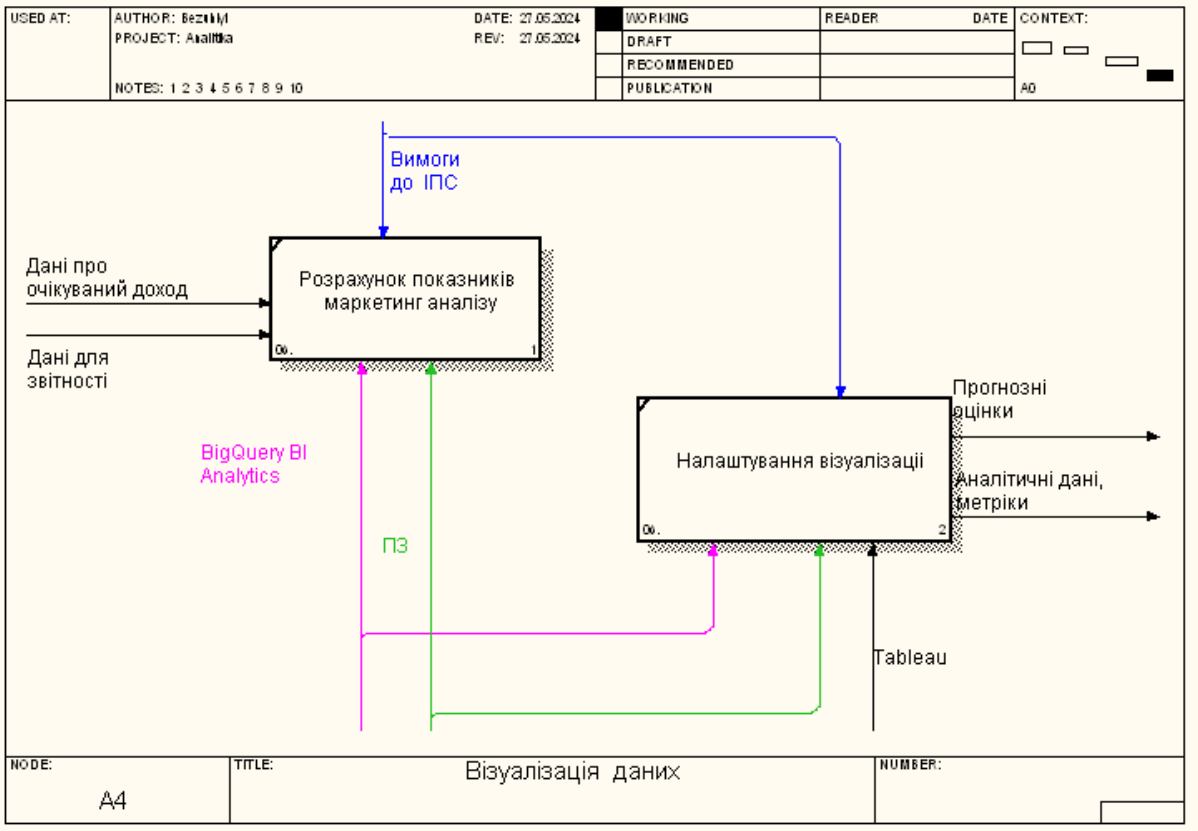


Рисунок 3.5 –Візуалізація результатів

Значення Refund - коефіцієнтів залежать від типу платежів (MASTERCARD/AMEX/ VISA/paypal-vault). На цьому етапі таблиці даних про поточний доход (Actuall revenue), значення Refund-коефіцієнтів, параметрів клієнтів та даних про їх місцезнаходження (Country groups) об'єднуються для обчислення прогнозованого доходу (Expected_revenue). Створюється окремо таблиця про всі види сплати за придбання продукту, де відображаються дані про всі види доходів (поточний та прогнозний) , дані про продукт, за який була зроблена передплата, всі параметри цієї передплати. На останньому кроці додаються дані про витрати на рекламу з Фейсбук для отримання фінальної таблиці для завантаження в Tableau.

Після розрахунку показників маркетингового аналізу та налаштування графічного та табличного подання результатів здійснюється в різних комбінаціях кількісне та графічне відображення за допомогою Tableau.

Отже, за результатами проведеного структурного аналізу було визначено основні етапи та процеси, які необхідні для досягнення мети роботи ІПС.

4 РОЗРОБКА ТА АНАЛІЗ МОДЕЛІ ДАНИХ

Інтегрована програмна система для Інтернет -маркетингу дозволяє здійснювати аналіз даних утримання або відтоку клієнтів, прогнозувати очікуваний доход, формувати звіт для розрахунку метрик маркетингової діяльності. Результати аналізу даних повинні бути надані в зрозумілій формі для експерта маркетолога. З цією метою за допомогою інструментальних засобів BI отримані результати необхідно візуалізувати для відображення показників маркетинг-аналізу [25]. На рисунку 3.1. надана структурна схема взаємозв'язку сервісів, що в використовуються для зберігання та обробки даних.

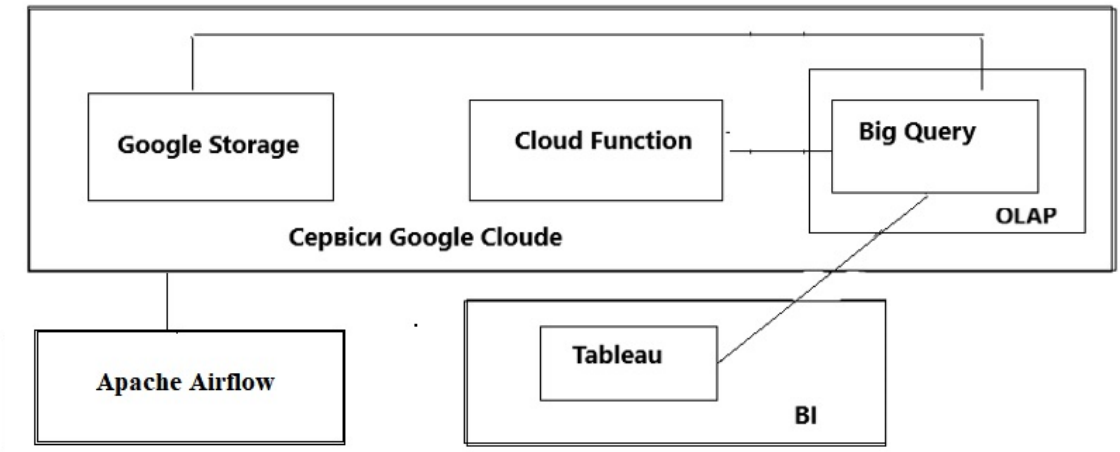


Рисунок 4.1 -Структурна схема взаємозв'язку сервісів зберігання та обробки даних

В хмарному сховищі Goooogle Storage зберігаються всі дані про користувачів та їх атрибуція (users attribution), дані про придбання продукту (subscription), дані про витрати на рекламу (insight), які надходять з Facebook Всі ці транзакцій дані завантажуються за кожний день впродовж 11 місяців. Структура та перелік полів таблиць наведено на рисунку 4.2.

Після попереднього очищення та обробки дані раз на добу зберігаються в Big Query. Всі операції з даними для відвантаження, завантаження, запуску аналітичних розрахунків виконуються за допомогою оркестратору.

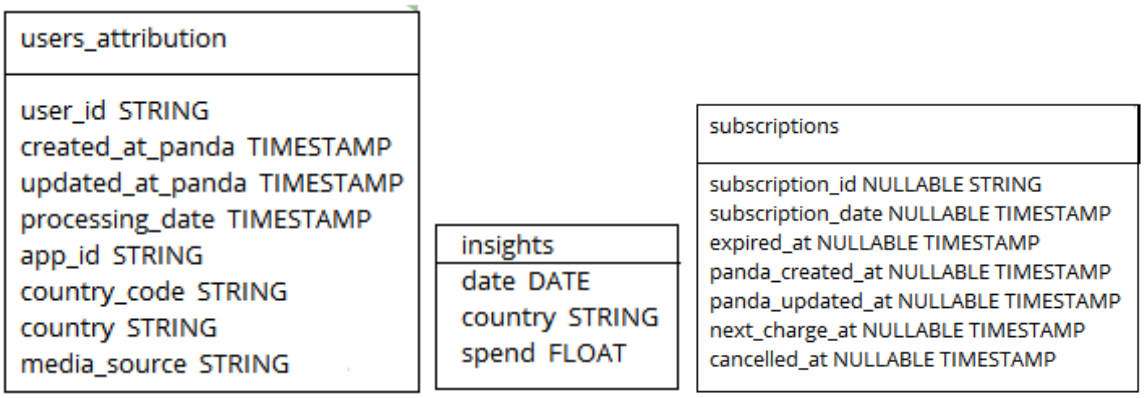


Рисунок 4.2. – Перелік полів таблиць транзакційних даних

Всі дані про зроблені передплати для придбання продукту зберігаються в таблиці Purchases, яка містить дані про продукт, завантажені дані про платежі по повному набору параметрів (наприклад, за передплату на пробний тріал період та передплату за підписку на певний термін), дані про користувачів, які очікують повернення коштів після їх списання. До набору параметрів також відносяться дані про країни походження покупок в тріал періоді та продовження передплати, що додані до таблиці Actual_Revenue. Схему агрегації даних в таблицю Actual_Revenue наведено на рисунку 4.3., поля таблиць представлені на рисунку 4.4

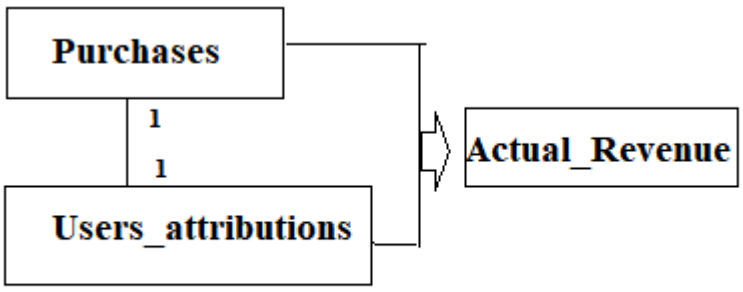


Рисунок 4.3. – Агрегація даних Actual_Revenue

Для розрахунку прогнозних оцінок очікуваного доходу потрібно розрахувати коефіцієнти утримання клієнтів Rebill-Coeff. Для їх розрахунку використовується модель sBG для кожної групи параметрів. З цією метою

потрібно виконати попередню обробку даних – вибрати користувачів у яких вже була сплачена передплата у визначеному для прогнозування періоді. Далі необхідно провести групування по даті платежу та по його номеру, групування по кластеру країн (Country_group), визначити кількість платежів в кожному розрізі параметрів.

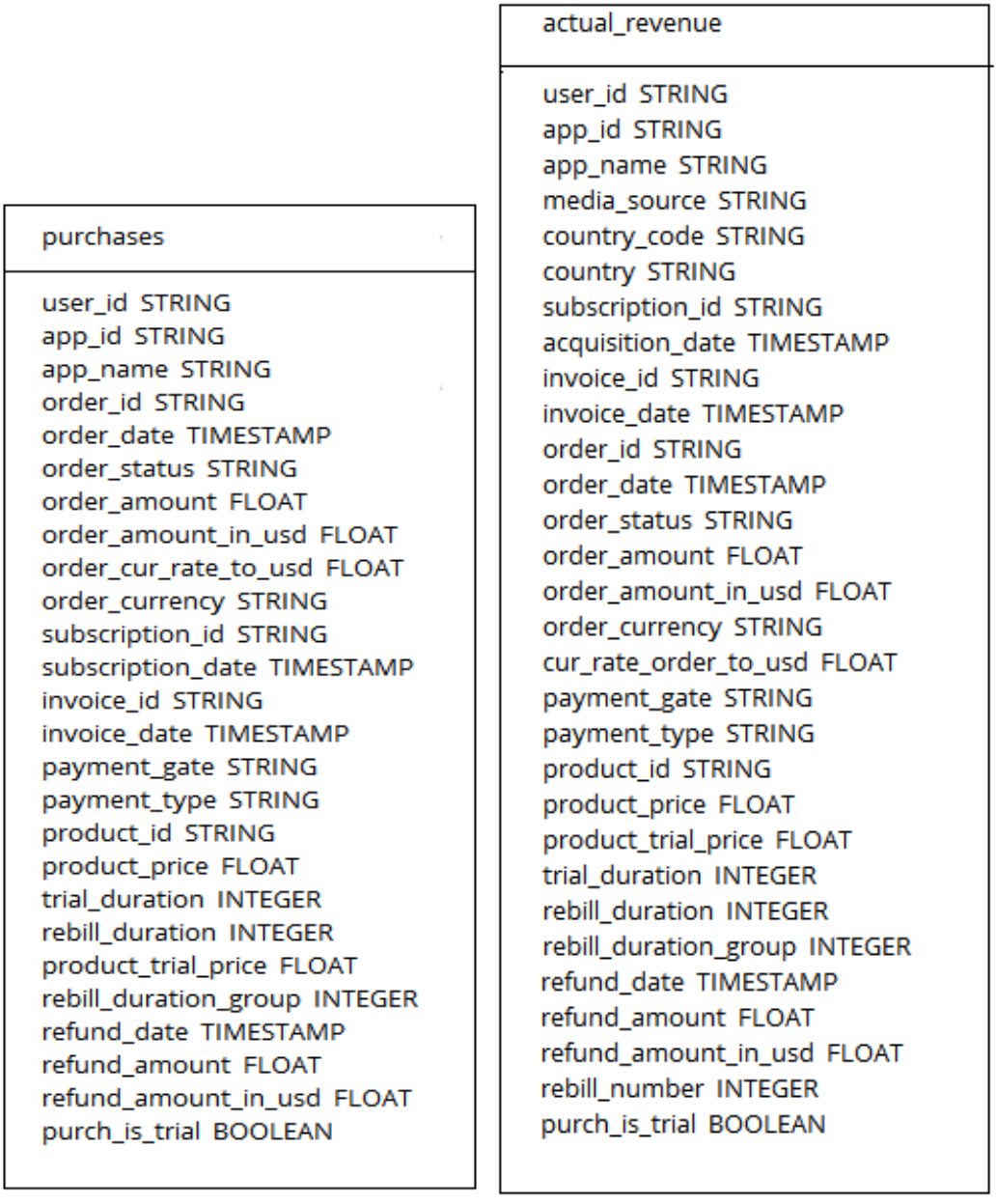


Рисунок 4.3. – Таблиці про зроблені передплати та поточний доход

При обчисленні коефіцієнтів використовується зменшення розмірності даних для визначеного періоду часу. Таким чином отримаємо список для

застосування розрахунку Rebill-Coeff, який виконується за допомогою Cloud Function, запуск розрахунку виконується автоматично за відправленням команд з оркестратору. В загальному вигляді на вхід моделі розрахунку коефіцієнтів подається список списків, де кожний внутрішній список – це одна когорта, тобто група користувачів за одної датою платежу, в якої в порядку спадання вказані значення кількості платежів для кожного номеру повторної передплати для кожного набору параметрів.

Результати розрахунків коефіцієнтів утримання клієнтів Rebill-Coeff додаються до даних поточного доходу. атрибуції клієнтів та створюється таблиця Expected_revenue шляхом агрегації даних, схему агрегація надано на рисунку 4.5.

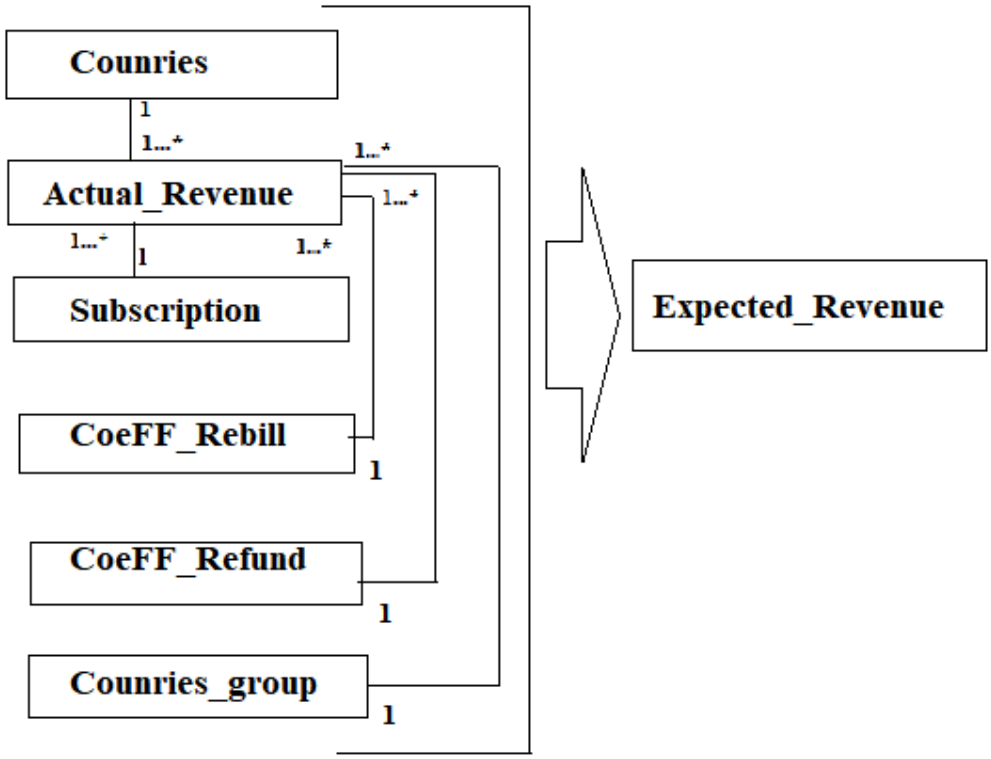


Рисунок 4.5. – Агрегація даних Expected_ Revenue

Прогнозні оцінки очікуваного доходу обчислюються на основі коефіцієнтів утримання клієнтів Rebill-Coeff. Розглянемо взаємодію даних для їх розрахунку, який виконується за допомогою моделі sBG для кожної групи параметрів. З цієї метою потрібно виконати попередню обробку даних – вибрати користувачів у яких вже була сплачена передплата у визначеному

для прогнозування періоді. Далі необхідно провести групування по даті платежу та по його номеру, групування по кластеру країн (Country_group), визначити кількість платежів в кожному розрізі параметрів. При обчисленні коефіцієнтів використовується зменшення розмірності даних для визначеного періоду часу. Таким чином отримаємо список для застосування розрахунку Rebill-Coeff, який виконується за допомогою Cloud Function, запуск розрахунку виконується автоматично за відправленням команд з оркестратору. В загальному вигляді на вхід моделі розрахунку коефіцієнтів подається список списків, де кожний внутрішній список – це одна когорта, тобто група користувачів за одної датою платежу, в якої в порядку спадання вказані значення кількості платежів для кожного номеру повторної передплати для кожного набору параметрів.

Приклади полів отриманої таблиці Expected_ Revenue на рисунку 4.6:

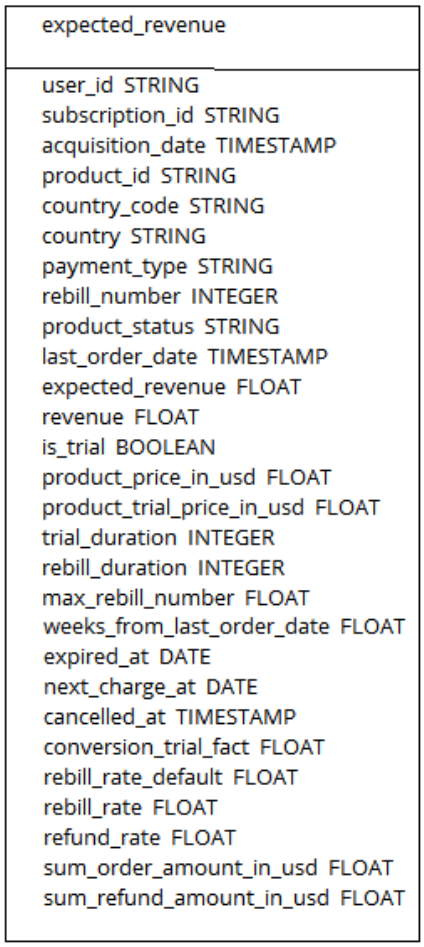


Рисунок 4.6.- Таблиця Expected_ Revenue

В додатку А наведено SQL код створення таблиці очікуваного доходу, побудованої на основі розрахованих прогнозних оцінок коефіцієнтів утримання клієнтів. Прогнозні оцінки доходу обчислюються з урахуванням значень коефіцієнтів утримання та відтоку клієнтів за формулою:

expected_revenue = (product_price_in_usd * rebill_rate)* (1 - refund_rate) - refund_rate * rebill_rate

revenue = sum_order_amount_in_usd - sum_refund_amount_in_usd

Для отримання даних про всі зроблені передплат та про всі очікувані оплати за підписку створюється таблиця Users_Purchases (дивись рисунок 4.7), яка об’єднує дані про фактичний та прогнозований доход (таблиці Expected_ Revenue, Actual Revenue), дані про підписки (Subscription) та країни (Countries).

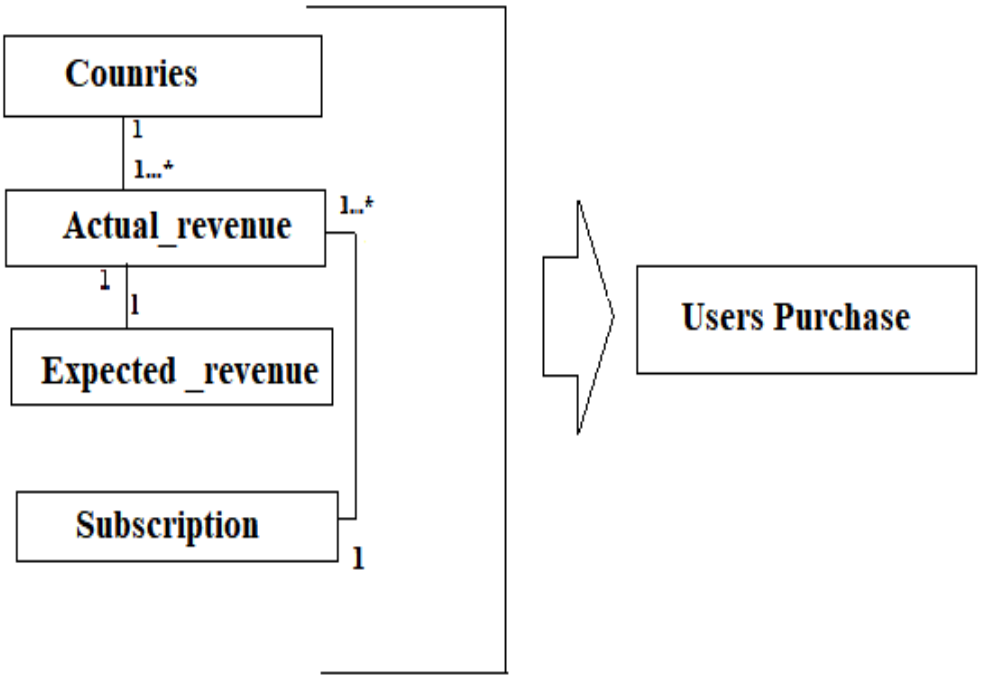


Рисунок 4.7. – Агрегація даних таблиця Users_Purchases

Остаточний звіт, який використовується в Tableau для візуалізації та обчислення метрик маркетингового аналізу містить дані про фактичний та прогнозований доход, дані про користувачів та країн їх походження, дані про продукт, який був передплачений користувачами. Для отримання звіту виконується агрегація даних, схема якої наведено на рисунку 4.7.

Створений звіт розміщується в проміжному файлі *.csv та завантажується в Tableau.

Увесь процес завантаження даних, послідовній агрегації таблиць для створення звіту, передача даних для обчислення коефіцієнтів утримання клієнтів виконується за допомогою оркестратору Apache Airflow. Створений оркестратором направлений ациклічний граф (DAG direct acyclic graph) містить розклад запуску певним чином визначених задач з метою конвеєризації обробки даних.

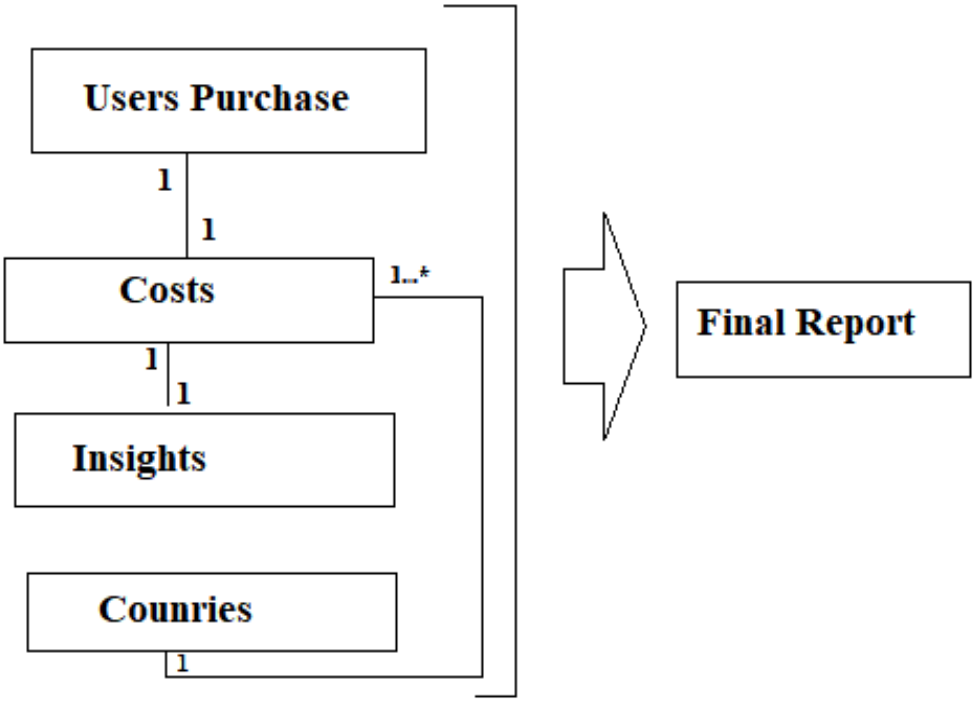


Рисунок 4.8. – Агрегація даних Final _Report

Отже в розділі розглянути всі типи даних, побудовані структури та послідовності обробки та агрегації даних з метою отримання даних щодо очікуваного доходу в розрізі до різних комбінацій параметрів в умовах конвеєрного їх завантаження та обробки згідно визначеного розкладу.

5 РОЗРОБЛЕННЯ АЛГОРИТМУ ФОРМУВАННЯ ПРОГНОЗНИХ ОЦІНОК ДОХОДУ

Для розрахунку прогнозних оцінок очікуваного доходу потрібно розрахувати коефіцієнти утримання клієнтів Rebill-Coeff за моделлю sBG . Розглянемо вхідні дані:

- initial_payment_date — дата першої покупки;
- product_id — id продукта;
- country_group — група краї, які були розділені на групу «0» (”краща” група), «1» (”гірша” група) та група «-1» (США);
- rebill_duration — атрибут підписки, далі потрібен для обчислень;
- trial_duration — атрибут підписки, далі потрібен для обчислень;
- rebill_number — номер повторного платіжу;
- payments — кількість покупок для кожної групи параметрів (product_id, country_group).

Перший крок. Завантаження необхідної платіжної інформації з BigQuery в Google Cloud Functions, де знаходиться код для обчислення прогнозних оцінок коефіцієнтів утримання клієнтів. Для завантаження даних використовується запит з файлу get_data_for_coeffs_rebills.sql (дивись додаток Б). Відкидаються тріальні транзакції (rebill_number > 0), фільтруються когорти користувачів з моменту відбору користувачів, у яких перший повторний платіж відбувся/мав відбутися в періоді до 300 днів (today - 300 days). Далі вже у функції ми виберемо тільки ті когорти, де є як мінімум 3 повторних платежу. Іншими словами, initial_payment_date = acquisition_date + trial_period (дата, коли відбувся/мав відбутися перший платіж);

Другий крок . Проведення попередньої обробки даних – обчислюється змінна diff_from_last за формулою:

(initial_payment_date - min(initial_payment_date)) // cohorts_duration.

Це поле потрібно для об’єднання когорт, щоб надавати більшу вагу “новішим” даним. Для того , щоб не враховувати повторні переплати, які не

всі користувачі в когорті мали можливість зробити потрібно підрахувати max_cohort_assumption_rebill_date — який є останньою датою конкретного номеру передплати для певної когорти (він має бути менший ніж сьогоднішня дата), для того, в конкретній когорті були тільки завершені повторні передплати.

Третій крок. Наступним кроком виконується алгоритм зменшення розмірності: в циклі групуємо по всім можливим комбінаціям параметрів (в нашому випадку parameter_combinations = [[product_id, country_group], [product_id]]) та за полями:

- rebill_number,
- max_cohort_assumption_rebill_date
- rebill_duration,
- trial_duration,
- initial_payment_date,

рахуємо по кожному розрізу кількість платежів.

Четвертий крок. На цьому кроці для кожної комбінації, наприклад для розрізу двох параметрів [[product_id, country_group] буде отримано комбінацію [SUB_WEEKLY, -1], формуємо список списків (fit_matrix), де кожний вкладений список – є когорта (група користувачів, об'єднаних за датою залучення протягом визначеного періоду), а кожне значення – кількість платежів для кожного номеру передплати. Такий список формується для всіх комбінацій наших параметрів, використовуючи когорти, в яких є більше двох повторних передплат, наприклад fit_matrix = [[323, 166, 99, 73, 49], [210, 112, 91, 61]].

П'ятий крок. Для кожного списку з попереднього кроку застосовуємо модель sBG (`sbg_model.fit(fit_matrix)`) отримуємо параметри моделі альфа і бета, які необхідні для прогнозування.

Шостий крок. Побудова прогнозних оцінок, період прогнозу складає 190 днів від першого платіжу. Використовуючи отримані параметри моделі альфа і бета, прогнозуємо Rebill_Coeffs, записуємо в загальну таблицю даних та

вказуємо вагу цього прогнозу. Якщо для цього прогнозу не було використовувано всіх параметрів, то додатково потрібно отримані прогнозні оцінки перетворити на рядки даних для всіх можливих комбінацій параметрів (дивись рисунок 5.1). В алгоритмі в залежності від кількості врахованих параметрів (product_id, country_group) значення вагового коефіцієнту може дорівнювати «0» (якщо Rebill_Coeffs обчислений з урахуванням обох параметрів) або дорівнює «-1» (якщо Rebill_Coeffs обчислений з урахуванням параметру product_id).

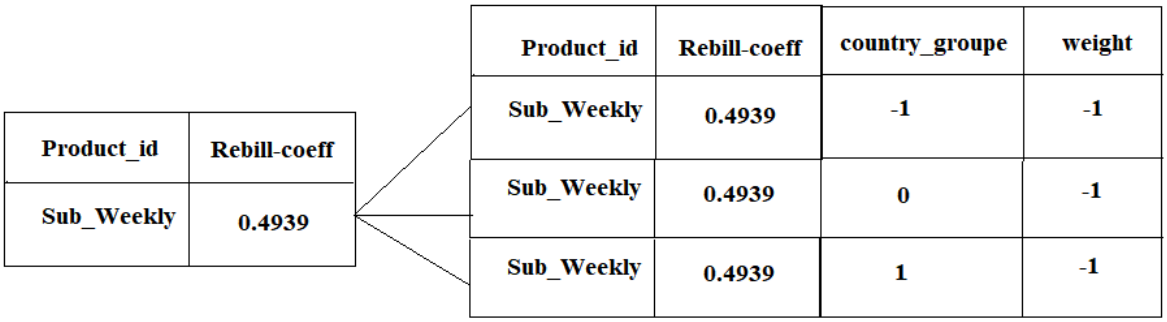


Рисунок 5.1. – Попереднє оброблення даних

На рисунку 5.2 наведено лістинг коду роботи обчислення прогнозних оцінок за методом sBG.

```
    # Initialize the sbg class
    model = sbg()

    # Fit the data to the model
    # data <- absolute number of payments for each cohort for each
rebill number
    data = [[323, 166, 99, 73, 49],  [210, 112, 91, 61]]
    model.fit(data)

    # Predict the survival probability for rebills
    survival_probabilities = model.predicted_survival(5)
    print(survival_probabilities)

    Output: [0.548, 0.369, 0.275, 0.218, 0.179]
```

Рисунок 5.2. – Лістинг розрахунку за моделлю sBG

Сьомий крок. . Вибираємо прогноз з найбільшим значенням weight для кожної комбінації параметрів. Отримані дані завантажуємо у таблицю BigQuery «Coeffs_rebill»

Розроблений алгоритм був реализований у програмний застосунок на мові програмування Python [26].

6 ВІЗУАЛІЗАЦІЯ ТА АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Для наочного подання даних в роботі використовується BI-інструмент, який не вимагає глибоких технічних знань. Tableau надає перевагу зручності для користувача, дозволяючи як технічним, так і нетехнічним користувачам створювати складні візуалізації та аналізи з легкістю. Він підтримує широкий спектр джерел даних, від електронних таблиць і баз даних до хмарних служб, забезпечуючи гнучкість і підключення.

На основі отриманих даних з фінального набору даних розраховані основні показники маркетингового аналізу, що дозволяють оцінити прибутковість вкладених в рекламу коштів та очікуваний доход (дивись рисунок 5.1).

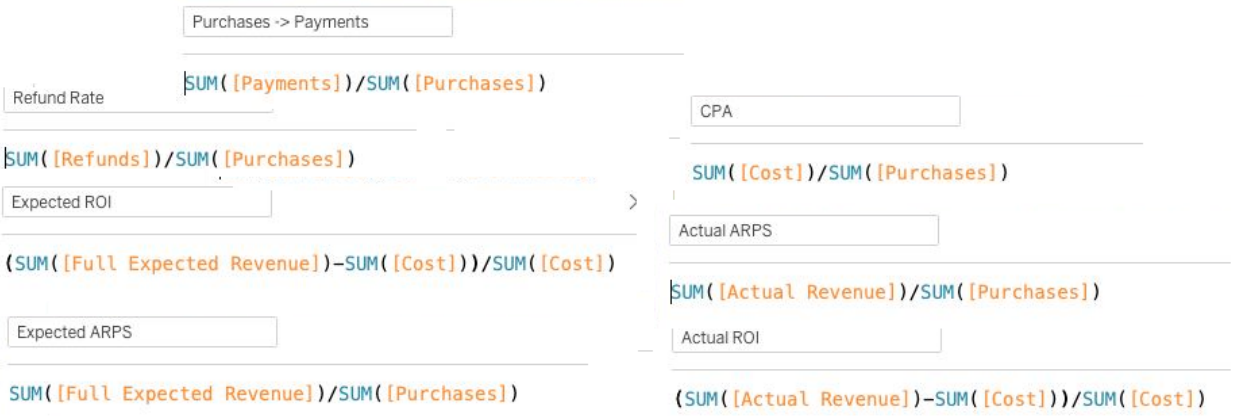


Рисунок 6.1 – Розрахунок показників МА

Загальний вигляд візуалізацій (дивись додаток Б) представлений двома типами звітів :

– головним звітом, який відображає всі кількісні розраховані значення як фактичні так й прогнозні за кожен день, графічне подання динаміки коефіцієнтів втрати та утримання клієнтів, гнучку систему фільтрації;

– звіту з поданням географічного відображення в розрізі гнучкого налаштування параметрів та ранжування за обраними критеріями.

Маркетологу або аналітику даних на головному звіті надана можливість вибирати країни, діапазон дат залучень клієнтів, налаштовувати дані відповідно до типів джерел надходження клієнта – органічні або з соціальних мереж

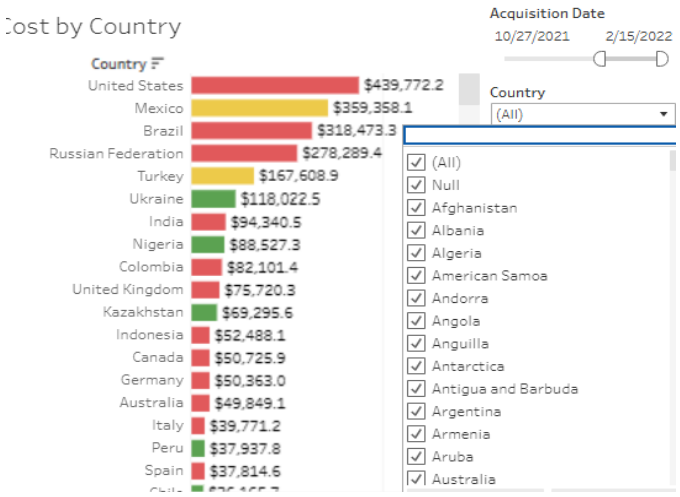


Рисунок 6.2 – Налаштування візуалізації за періодом часу та країною

Відображення кількісних значень показників МА для користувачів з соціальних мережі Facebook для всіх країн представлено на рисунку 5.3.

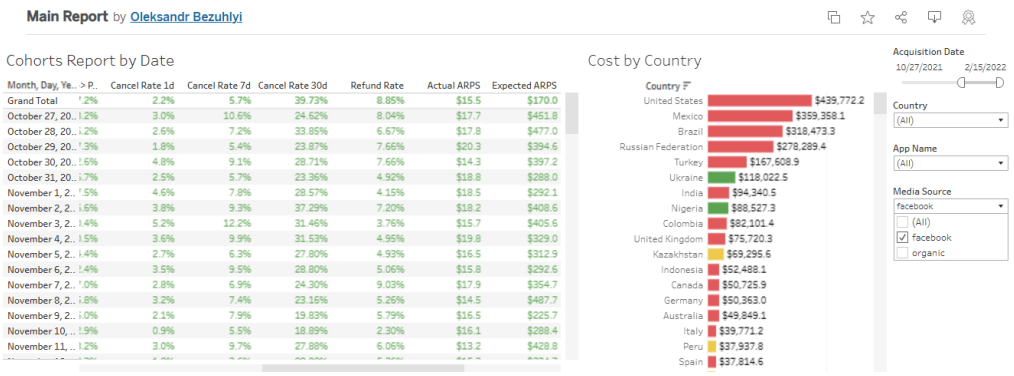


Рисунок 6.3 – Налаштування візуалізації за кількісними значеннями показників МА та джерелом надходження клієнта

Відповідні порівняльні графіки динаміки показників CPA (ціна за дію-cost-per-action) та очікуваної окупність інвестицій (ROI – return on investment) наведені на рисунку 6.4.

Чим нижче сума, котру рекламодавець сплачує за цільову дію користувача: покупку, реєстрацію, підписку, тим вигіднішою є рекламна кампанія Окупність інвестицій (ROI) відображає відношення прибутку від виконаних дій до витрат на них та відіграє важливу роль для подальшого вибору найвигідніших каналів рекламного трафіку чи способів збуту продукції.

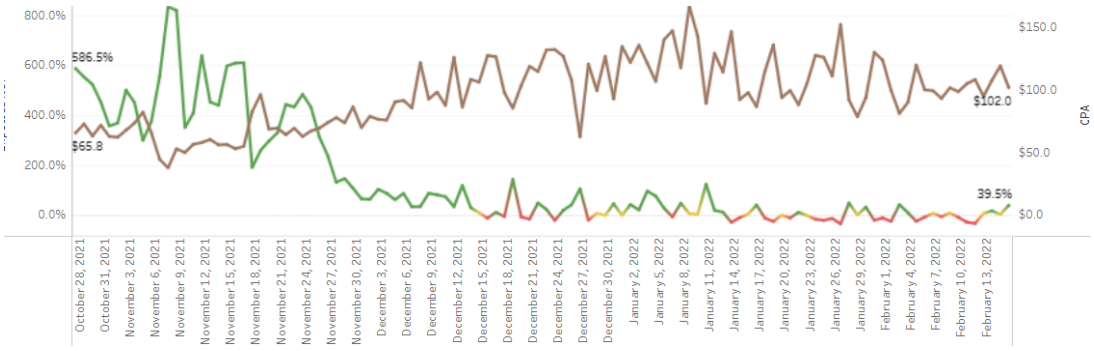


Рисунок 6.4 – Графічне відображення динаміки показників CPA та ROI

На рисунках 6.5-6.6 відображено графічне подання по країнах для даних отриманих з соціальної мережі та органічні дані, тобто отримані з сайту.

Рисунок 6.5 – Графічне відображення динаміки показників CPA та ROI

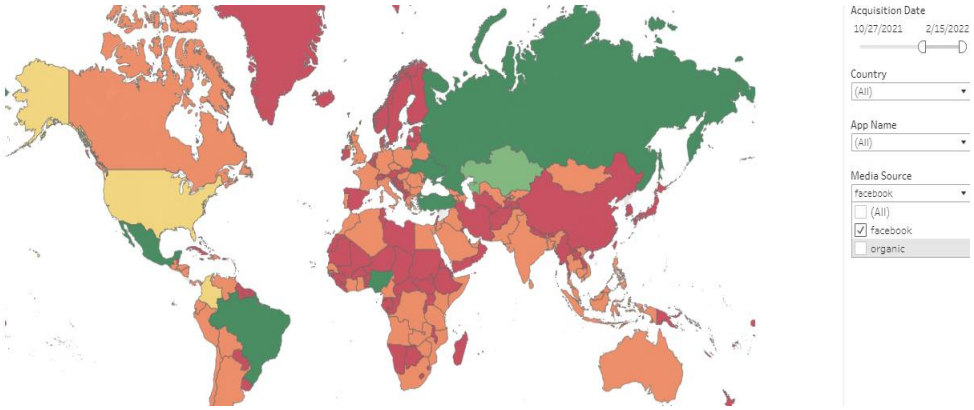


Рисунок 6.6 – Графічне відображення динаміки показників CPA та ROI

Tableau самостійно знайде географічні дані, або це можна зробити вручну. Сервіс запропонує побудувати картку та автоматично виділить контури країн, які вказані у даних. Колір країни на графіку залежить від обсягу продажів. Якщо потрібна детальна інформація по кожному місяцю по всім показникам МА та підсумкові дані для окремої країні, тоді потрібно навести курсор на обрану країну (дивись рисунок 6.7)

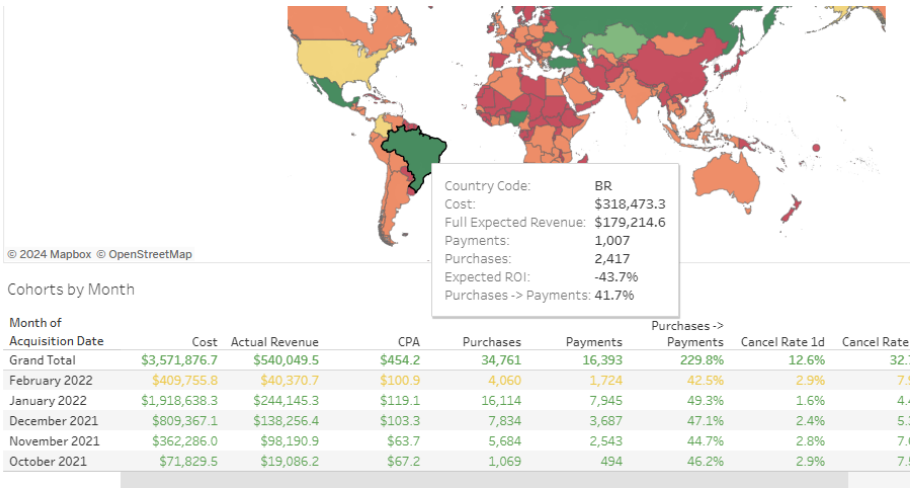


Рисунок 6.7 – Географічне відображення показників МА для окремої країни

Інформаційна панель Tableau надає користувачам цілісне та одночасне уявлення про декілька наборів даних. Аркуші та інформаційна панель пов’язані, а це означає, що якщо ви змінюєте аркуш, відповідна інформаційна панель змінюється, і навпаки. Обидва автоматично оновлюються, коли змінюється джерело даних.

ВИСНОВКИ

В кваліфікаційної роботі розроблено інтегровану програмну систему аналітики для Інтернет-маркетингу, яка реалізує алгоритм прогнозування показників очікуваного прибутку, з можливістю візуалізації визначених показників маркетингового аналізу даних за допомогою конвеєра даних для кампаній, метою яких є надання SaaS або DaaS послуг за умови оформлення придбання ресурсу або підписки на визначений термін. Задача, що розв'язана у робіти є складовою комплексної задачі – розроблення інтегрованої програмної системи аналітики для Інтернет-маркетингу для даних, які отримані в соціальній мережі Facebook за достатньо тривалий час з щоденним їх оновленням, дані про повторні передплати на онлайн сервіси в їх динаміці.

Досліджено особливості застосування інструментів бізнес-аналітики: технології сховищ даних, OLAP Big Query, OLTP, Tableau, Apache Airflow для розв'язку задачі прогнозування коефіцієнтів утримання клієнтів та відповідно очікуваного доходу з урахуванням витрат на рекламу.

За результатами аналізу математичних моделей для прогнозування коефіцієнтів утримання клієнтів було обґрунтовано застосування моделі sBG, проведено оцінка точності в отриманих прогнозних значень.

Розроблено та проаналізовано модель та структура даних во взаємозв'язку їх обробки за допомогою хмарних сервісів та оркестратору Apache Airflow , розроблена програмне забезпечення (SQL запит) для створення набору даних з обчисленими показниками маркетингу та прогнозними оцінками доходу , коефіцієнтів утримання клієнтів. Розроблено алгоритм, який реалізований програмно на Python для обчислення прогнозних оцінок за методом sBG,

Налаштовано інструмент Tableau для гнучкого відображення результатів у графічному та географічному поданні з урахуванням параметрів щодо дати залучення клієнтів, місця знаходження, кількісті передплат, джерела надходження даних з можливостю зробити порівняльний аналіз.

ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. Присакар І.І. Інтернет-маркетинг як сучасна платформа для розвитку бізнесу. Бізнес Інформ. 2015. № 12. С.333–339
2. Вдовічена, О., Гой, В., & Харів, В. (2023). ІНТЕРНЕТ МАРКЕТИНГ ЯК ЗАСІБ ПРОСУВАННЯ БРЕНДУ. Економіка та суспільство, (53). https://doi.org/10.32782/2524-0072/2023-53-51
3. BÜCHNER, Alex G.; MULVENNA, Maurice D. Discovering internet marketing intelligence through online analytical web usage mining. ACM Sigmod Record, 1998, 27.4: 54-61.
4. ВАВРИК, А. Б. Методи та інструменти інтернет-маркетингу. 2018. https://dspace.uzhnu.edu.ua/jspui/handle/lib/24252
5. IVANOV, Mykola. Cloud-based Digital Marketing. In M3E2-MLPEED. 2019. p. 395-404.
6. ОКЛАНДЕР, М. А.; РОМАНЕНКО, О. О. Специфічні відмінності цифрового маркетингу від інтернет-маркетингу. Економічний вісник Національного технічного університету України Київський політехнічний інститут, 2015, 12: 362-371
7. Сидорова А. В., Біленко Д. В., Буркіна Н. В. С 347 Бізнес-аналітика: навчально-методичний посібник. Вінниця: ДонНУ імені Василя Стуса. 2019. 104 с
8. A Spreadsheet-Literate Non-Statistician’s Guide to the Beta-Geometric Model https://www.brucehardie.com/notes/032/BG_intro.pdf
9. СУЧАСНІ ІНСТРУМЕНТИ ЦИФРОВОГО МАРКЕТИНГУ. ЖУРНАЛІСТИКА ТА РЕКЛАМА: ВЕКТОРИ ВЗАЄМОДІЇ. 2019. URL: https://doi.org/10.31617/k.knute.2019-03-19.74 (дата звернення: 06.06.2024).]
10. What Is Data-as-a-Service// [Електронний ресурс]. – Режим доступу: https://builtin.com/articles/data-as-a-service-daas (дата звернення 04.06.2024)

11. What is SaaS (Software as a Service)? – Режим доступу: https://aws.amazon.com/what-is/saas/ (дата звернення 04.06.2024)
12.. Cost Per Action (CPA) // [Електронний ресурс]. – Режим доступу: https://horoshop.ua/ua/glossary/cost-per-action/(дата звернення 02.06.2024
13. APRA (Average Revenue Per Account) // [Електронний ресурс]. – Режим доступу: https://sok.marketing/apra//(дата звернення 02.06.2024
14. Churn Rate: What It Means, Examples, and Calculations // Режим доступу: https://www.investopedia.com/terms/c/churnrate.asp (дата звернення 02.06.2024)
15. What Is the Refund Rate? // [Електронний ресурс]. – Режим доступу: https://seon.io/resources/dictionary/refund-rate/ (дата звернення 04.06.2024)
16. 10 ключових показників у retention-маркетингу// [Електронний ресурс]. – Режим доступу: https://esputnik.com/blog/10-klyuchevyh-pokazatelei-v-retention-marketinge (дата звернення 04.06.2024)
17. What is Tableau - The Complete Guide to Tableau// [Електронний ресурс]. – Режим доступу: https:// https://www.datacamp.com/blog/all-about-tableau (дата звернення 01.06.2024)
18. Become an AI Enterprise: Experience the power of a personalized Tableau Conference to You now! // [Електронний ресурс]. – Режим доступу: https:// https://www.tableau.com (дата звернення 01.06.2024)
19. Peter S. Fader com Bruce G. S. Hardie A Spreadsheet-Literate Non-Statistician's Guide to the Beta-Geometric Model www.petefader.com, 2014
20. Calculating Customer Lifetime Values using a Shifted-Beta-Geometric model// [Електронний ресурс]. – Режим доступу: https:// https://towardsdatascience.com/calculating-customer-lifetime-values-using-a-shifted-beta-geometric-model-86bf538444f4 (дата звернення 01.06.2024)
21. Fader, Peter S. and Bruce G. S. Hardie (2007a), "How to Project Customer Retention," Journal of Interactive Marketing, 21 (Winter), 76–90.

22. Fader, P. S., & Hardie, B. G. (2007). Fitting the sBG model to multi-cohort data.
23. Аналіз великих даних у фінансах / Big Data Analytics in Finance https://learn.ztu.edu.ua/pluginfile.php/253060/mod_resource/content/
24. Мінухін С. В., Беседовський О. М., Знахур С. В. Методи і моделі проектування на основі сучасних CASE-засобів : навч. посіб. Харків : Вид. ХНЕУ, 2008. 272 с.
25. Dimensional Modeling:In a Business Intelligence Environment https://www.redbooks.ibm.com/redbooks/pdfs/sg247138.pdf
26. Predictive Modeling w/ Python. https://medium.com/@ben.putney/list/predictive-modeling-w-python-e3668ea008e1
27. ДСТУ 3008:2015 — «Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання»— державний стандарт України

ДОДАТОК А
SQL Код створення таблиці даних з результатами розрахунку «Expected Revenue»

```
CREATE OR REPLACE TABLE final.expected_revenue AS (

WITH

subs AS (
    SELECT
        subscription_id          AS subscription_id
        , DATE(expired_at)       AS expired_at
        , DATE(next_charge_at)   AS next_charge_at
        , cancelled_at           AS cancelled_at
    FROM exodus.subscriptions
),

refund_coeffs AS (
    SELECT DISTINCT
        payment_type                    AS payment_type_refund
        , country_cluster               AS country_cluster_refund
        , rebill_number                 AS rebill_number
        , rebill_duration_group         AS rebill_duration_group
        , refund_rate                   AS refund_rate
        , refund_rate_residual          AS refund_rate_residual
    FROM final.coeffs_refund
),

refund_countries AS (
    SELECT DISTINCT country_code,
                    refund_group        AS country_cluster
    FROM exodus.countries_groups
),

trial_rebill_countries AS
(
    SELECT country_code,
           trial_rebill_group           AS country_group_trial_rebill
    FROM exodus.countries_groups
),

-- last_created_at_rebills AS (
--     SELECT DISTINCT created_at
--     FROM `final.coeffs_rebill`
--     ORDER BY created_at DESC
--     LIMIT 5
-- ),

rebill_coeffs_raw AS (
```

```
    SELECT
        product_id
AS product_id
        , country_group
AS country_group_trial_rebill
        , rebill_number
AS rebill_number
        , AVG(rebill_rate)
AS rebill_rate
        , 1
AS weight

    FROM `final.coeffs_rebill`

        -- RIGHT JOIN last_created_at_rebills
        -- USING(created_at)

    GROUP BY
      product_id
      , country_group
      , rebill_number

),

rebill_coeffs AS (
    SELECT
        product_id
        , country_group_trial_rebill
        , rebill_number
        , rebill_rate
        , FIRST_VALUE(rebill_rate)
            OVER (PARTITION BY product_id,
                               country_group_trial_rebill
                  ORDER BY rebill_number, weight DESC)
AS rebill_rate_default

    FROM rebill_coeffs_raw

    QUALIFY ROW_NUMBER() OVER (PARTITION BY product_id,

country_group_trial_rebill, rebill_number
                               ORDER BY weight DESC) = 1
),

base_users_payment_data AS (
    SELECT
        user_id
AS user_id
    -- since upsels don't have subscription_id, we add order_id to use
it as identifier
```

```
        , COALESCE(subscription_id, order_id)
AS subscription_id
        , acquisition_date
AS acquisition_date

        , product_id
AS product_id
        , country_code
AS country_code
    -- define the platform values as IOS and Android - everything else
is thrown into Android
    -- this is necessary to calculate the conversion to trials and
rebill rate
        , payment_type
AS payment_type
        , COALESCE(product_price, 0)
AS product_price_in_usd
        , COALESCE(product_trial_price, 0)
AS product_trial_price_in_usd

        , COALESCE(trial_duration, 0)
AS trial_duration
        , COALESCE(rebill_duration, 0)
AS rebill_duration
        , rebill_duration_group
AS rebill_duration_group
        , MAX(order_date)
AS last_order_date
    -- define the rebill_number for upsels as -1
    -- this will then be used to join the coefficients
        , rebill_number
AS rebill_number

        , SUM(order_amount_in_usd)
AS sum_order_amount_in_usd

        , COUNT(IF(rebill_number=0, NULL, refund_date))
AS number_of_refunds
        , SUM(COALESCE(refund_amount_in_usd, 0))
AS sum_refund_amount_in_usd

    FROM final.actual_pyrche

    WHERE acquisition_date >= CAST(DATE_SUB('2022-02-01', INTERVAL 220
DAY) AS TIMESTAMP)
            AND product_id IS NOT NULL

    GROUP BY
        user_id
        , COALESCE(subscription_id, order_id)
        , acquisition_date
```

```
        , product_id
        , country_code
        , payment_type
        , COALESCE(product_price, 0)
        , COALESCE(product_trial_price, 0)
        , COALESCE(trial_duration, 0)
        , COALESCE(rebill_duration, 0)
        , rebill_duration_group
        , rebill_number

),
processed_users_payment_data AS (
    SELECT
        base.user_id
        , base.subscription_id

        , base.acquisition_date
        , base.product_id
        , base.country_code
        , base.payment_type
        , base.product_price_in_usd
        , base.product_trial_price_in_usd
        , base.trial_duration
        , base.rebill_duration

        , base.rebill_duration_group
        , base.last_order_date
        , base.rebill_number                                    AS
rebill_number
        , base.sum_order_amount_in_usd
        , base.number_of_refunds
        , base.sum_refund_amount_in_usd

        , subs.expired_at                                       AS
expired_at
        , subs.next_charge_at                                   AS
next_charge_at
        , subs.cancelled_at                                     AS
cancelled_at

        , IF(base.rebill_number = 0, TRUE, FALSE)               AS
is_trial

        , IF(base.rebill_duration > 0,
                        IF(190 < base.trial_duration, 0, CEIL((190 -
base.trial_duration) / base.rebill_duration)),
                        -1)
AS max_rebill_number
        , FLOOR(
           DATE_DIFF(CURRENT_TIMESTAMP(),
```

```
                    base.last_order_date, DAY) / 7)
AS weeks_from_last_order_date

        , IF(
            base.payment_type NOT IN (
                SELECT DISTINCT payment_type_refund FROM
refund_coeffs),
           'Other', base.payment_type
          )
AS payment_type_refund
        , COALESCE(trial_rebill_countries.country_group_trial_rebill,
1)    AS country_group_trial_rebill
        , COALESCE(refund_countries.country_cluster, 0)
AS country_cluster_refund

  FROM base_users_payment_data AS base
          LEFT JOIN refund_countries USING (country_code)
          LEFT JOIN trial_rebill_countries USING (country_code)
          LEFT JOIN subs USING (subscription_id)
 ),

 processed_users_payment_data_with_coeffs AS (
    SELECT
        proc_base.user_id
        , proc_base.subscription_id

        , proc_base.acquisition_date
        , proc_base.product_id
        , proc_base.country_code
        , proc_base.payment_type

        , proc_base.product_price_in_usd
        , proc_base.product_trial_price_in_usd
        , proc_base.trial_duration
        , proc_base.rebill_duration
        , proc_base.rebill_duration_group

        , proc_base.last_order_date
        , proc_base.rebill_number
        , proc_base.sum_order_amount_in_usd
        , proc_base.number_of_refunds
        , proc_base.sum_refund_amount_in_usd

        , proc_base.expired_at
        , proc_base.next_charge_at
        , proc_base.cancelled_at

        , proc_base.is_trial
        , proc_base.max_rebill_number
        , proc_base.weeks_from_last_order_date
```

```
        , proc_base.payment_type_refund
        , proc_base.country_group_trial_rebill
        , proc_base.country_cluster_refund

        , COUNT(
            IF(proc_base.rebill_number > 0, proc_base.user_id, NULL)
         ) OVER (window_for_conv_tr_rb)
         /
         COUNT(*) OVER (window_for_conv_tr_rb)
AS conversion_trial_fact


        , rebill_coeffs.rebill_rate
AS rebill_rate

        , CASE

              WHEN proc_base.rebill_number >=
proc_base.max_rebill_number
                  THEN 0
              WHEN rebill_coeffs.rebill_rate IS NULL
                  THEN 0
              ELSE rebill_coeffs.rebill_rate_default
          END
AS rebill_rate_default


        , CASE
              WHEN proc_base.number_of_refunds = 0
                  THEN IF(proc_base.rebill_number >=
proc_base.max_rebill_number,
                          0, refund_coeffs.refund_rate)

              WHEN proc_base.number_of_refunds > 0
                  THEN 0

              ELSE IF(proc_base.rebill_number >=
proc_base.max_rebill_number + 1,
                      0, refund_coeffs.refund_rate_residual)
          END
AS refund_rate

        , IF(proc_base.payment_type = 'paypal-vault',
                0,
                IF(proc_base.rebill_number=-1, 0.3, 0.2)
        )
AS prevention_ratio

        , COUNT(*) OVER (window_for_conv_tr_rb)
AS all_users_for_adjstmt
```

```
        , SUM(proc_base.rebill_number)
            OVER (window_for_conv_tr_rb)
AS sum_act_rebills_for_rb_adjstmt

    FROM processed_users_payment_data AS proc_base

       LEFT JOIN rebill_coeffs
      USING (product_id, country_group_trial_rebill, rebill_number)

       LEFT JOIN refund_coeffs
      USING (rebill_duration_group, payment_type_refund,
country_cluster_refund, rebill_number)

    WINDOW
         window_for_conv_tr_rb AS (PARTITION BY
DATE(acquisition_date), product_id, country_group_trial_rebill)
)

SELECT
    user_id
    , subscription_id
    , acquisition_date
    , product_id
    , COALESCE(country_code, 'Not Defined')                    AS
country_code
    , COALESCE(countries.name, 'Not Defined')                  AS
country
    , payment_type

    , rebill_number
    , last_order_date

    , (
        (product_price_in_usd * rebill_rate)
            * (1 - refund_rate)

        - refund_rate * rebill_rate
    )                                                          AS
expected_revenue

    , (
        sum_order_amount_in_usd
            - sum_refund_amount_in_usd
    )                                                          AS
revenue

    , is_trial
    , product_price_in_usd
    , product_trial_price_in_usd
    , trial_duration
```

```
    , rebill_duration
    , max_rebill_number
    , weeks_from_last_order_date
    , expired_at
    , next_charge_at
    , cancelled_at
    , conversion_trial_fact
    , rebill_rate_default
    , rebill_rate
    , refund_rate
    , sum_order_amount_in_usd
    , sum_refund_amount_in_usd
    , rebill_duration_group
    , country_group_trial_rebill
   , country_cluster_refund
  FROM processed_users_payment_data_with_coeffs AS base
      LEFT JOIN exodus.countries AS countries
          ON countries.code = base.country_code
  QUALIFY ROW_NUMBER() OVER (PARTITION BY user_id, product_id,
subscription_id ORDER BY rebill_number DESC) = 1
)
```

Додаток Б Текст програми вибору даних для розрахунку коефіцієнтів

```
WITH rebills_prep AS (
  SELECT
      product_id
AS product_id
-- make assumptions about the date of the first rebill
    , DATE_ADD(acquisition_date,INTERVAL COALESCE(trial_duration, 0)
DAY) AS initial_payment_date
    , CASE WHEN trial_rebill_group IS NOT NULL THEN trial_rebill_group
        ELSE 1 END
AS country_group
    , rebill_duration
AS rebill_duration
    , COALESCE(trial_duration, 0)
AS trial_duration
    , rebill_number
AS rebill_number
    , user_id
AS user_id
    FROM final.actual_revenue AS ar

        LEFT JOIN exodus.countries_groups cg
            ON cg.country_code = ar.country_code

   WHERE rebill_duration BETWEEN 1 AND 365
     AND rebill_number > 0
     AND DATE_ADD(acquisition_date, INTERVAL COALESCE(trial_duration,
0) DAY)
         >= CAST(DATE_SUB(DATE('{date}'), INTERVAL 300 DAY) AS
TIMESTAMP)
  )

SELECT
    initial_payment_date
  , product_id
  , country_group
  , rebill_duration
  , trial_duration
  , rebill_number
  , COUNT(DISTINCT user_id) AS payments

  FROM rebills_prep

 GROUP BY
    initial_payment_date
  , product_id
  , country_group
  , rebill_duration
  , trial_duration
  , rebill_number;
```

ДОДАТОК В

Текст програми обчислення прогнозних оцінок Rebill_coeffs

```
from typing import List, Dict
import pandas as pd
import datetime
from typing import List
import numpy as np
from scipy.optimize import minimize
from utils.google_cloud_services.cloud_functions_logging import Logger
from utils.google_cloud_services.bigquery import
(upload_data_from_bq_into_dataframe,

insert_dataframe_into_bq_table)


def survivor(probabilities, t) -> float:
    """
    Survivor function of the sBG distribution
    """
    s = 1 - probabilities[0]
    for x in range(1, t + 1):
        s = s - probabilities[x]
    return s


def probability(alpha: float, beta: float, t: int) -> float:
    """
    Probability mass function of the sBG distribution
    """
    if t == 0:
        return alpha / (alpha + beta)
    return (beta + t - 1) / (alpha + beta + t) * probability(alpha,
beta, t - 1)


def log_likelihood_multi_cohort(alpha: float, beta: float, data:
List[list]) -> float:
    """
    Function to maximize to obtain ideal alpha and beta parameters
        using data across multiple (contiguous) cohorts.

    :param data: must be a list of lists with cohorts each with an
absolute number
            of rebills per observed time unit
    :return: the likelihood of observing the data at certain alpha and
beta
    """
    if alpha <= 0 or beta <= 0:
        return -1000

    probabilities_size = max(len(data[0]), len(data))
```

```
    probabilities = [probability(alpha, beta, i) for i in
range(probabilities_size)]

    cohorts = len(data)
    total = 0
    for i, cohort in enumerate(data):
        total += sum([(cohort[j] - cohort[j + 1]) *
np.log(probabilities[j]) for j in range(len(cohort) - 1)])
        total += cohort[-1] * np.log(survivor(probabilities, cohorts -
i - 1))
    return total


def predicted_retention(alpha: float, beta: float, t: int) -> float:
    """
    Predicted retention probability at t
    """
    return (beta + t) / (alpha + beta + t)


class sbg:
    '''
    A class used for fitting and predicting rebill rates
        using sBG distribution

    Attributes
    ----------
    alpha: float
        parameter of the sBG distribution
    beta: float
        parameter of the sBG distribution
    success_flag: bool
        a flag indicating that the distribution parameters were
successfully found
    '''

    def __init__(self):
        self.alpha = None
        self.beta = None
        self.success_flag = True

    def fit(self, data: List[list]) -> None:
        """
        Maximize the likelihood of observing the data given some
parametric setting
            (alpha and beta) and define them
        Since вЂњminimizing lossвЂќ makes more sense,
            we can instead take the negative of the log-likelihood and
minimize that
```

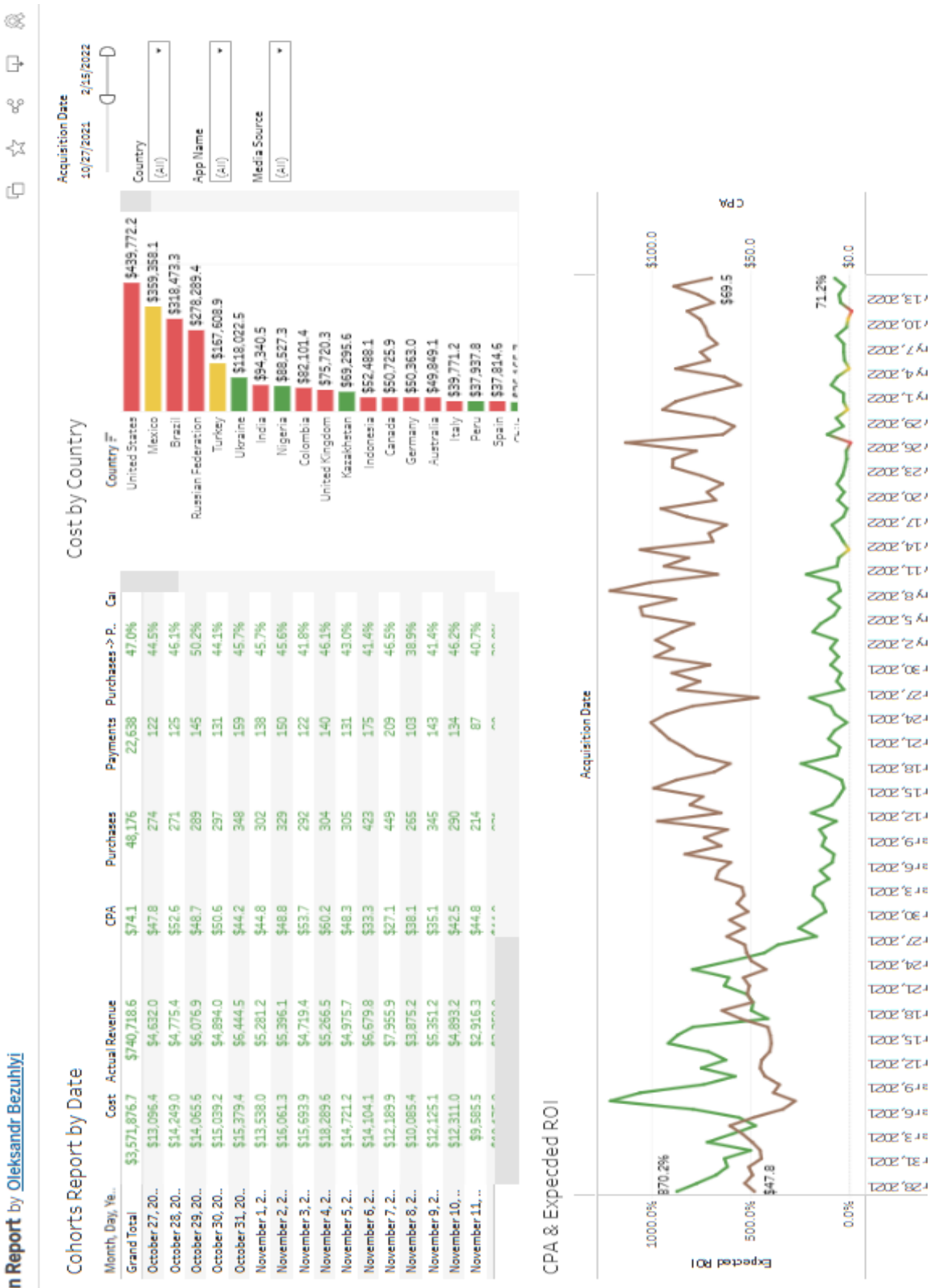
```
        :param data: must be a list of lists with cohorts each with an
absolute number
            of rebills per observed time unit
        """
        func = lambda x: -log_likelihood_multi_cohort(x[0], x[1],
data)
        x0 = np.array([1., 1.])

        try:
            res = minimize(func, x0, method='nelder-mead')
            self.alpha, self.beta = res.x
        except Exception as err:
            print(err)
            self.success_flag = False

    def predicted_survival(self, x: int) -> list:
        """
        Predicted survival probability, i.e. percentage of customers
retained, for all t in x

        :param x: the number of rebills for the forecast
        :return: list with retention forecast for each rebill number
        """
        s = [predicted_retention(self.alpha, self.beta, 0)]

        for t in range(1, x):
            s.append(predicted_retention(self.alpha, self.beta, t) *
s[t - 1])
        return [1] + s


def handle_fit_matrix(data: List[list]) -> List[list]:
    """
    Processes data to the form required for fitting:
        drop null values & complements data of the first cohort
        (the amount of data in the first cohort should be the largest
-
        it is necessary for the algorithm to work properly)

    :param data: list of lists with cohorts with the amount of
payments
    :return: processed list of lists with cohorts with the amount of
payments
    """
    data = [[j for j in i if ~np.isnan(j)] for i in data]

    max_cohort_size = max([len(i) for i in data])
    if max_cohort_size == len(data[0]):
        return data
    else:
        data[0].extend([0 for _ in range(max_cohort_size)])
```

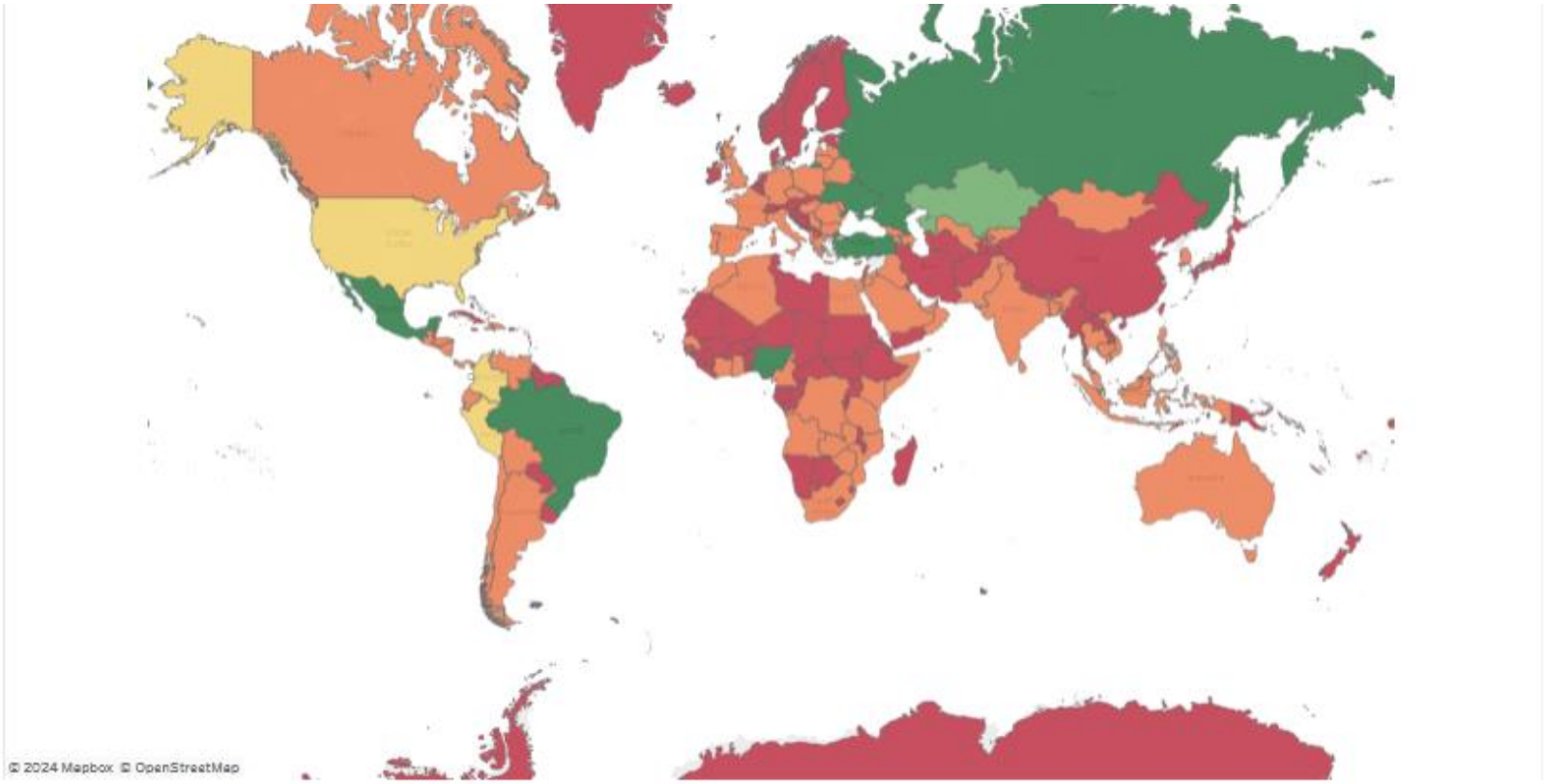
```
        data[0] = [data[0][i] for i in range(max_cohort_size)]
        return data


def expected_coeffs_rebills_calc(query_path_for_data_sampling: str
                                 , execution_date: str
                                 , params: list
                                 , table_to_which_insert: str
                                 , trace: str
                                 , format_info: Dict = {}
                                 , total_payments_threshold=30
                                 , retry_window=10
                                 , cohorts_duration=20) -> None:
    """
    Calculates and insert to BigQuery table expected rebill rates for
forecast ARPS

    :param query_path_for_data_sampling:
        path to the file where the query for data selection is stored
    :param params: parameters of the groups for combined forecast
    :param total_payments_threshold: the minimum number of
total_payments required
        for a cohort to be selected for prediction
    :param retry_window: the number of days that passed after the
first retry attempt
        required for the cohort to be selected for prediction
    :param cohorts_duration: number of days to divide users into
cohorts
    """

    logging = Logger(trace)

    execution_date = datetime.datetime.strptime(execution_date, '%Y-
%m-%d').date()

    rebills =
upload_data_from_bq_into_dataframe(query_filepath=query_path_for_data_
sampling, format_info=format_info)
    logging.debug("Successfully uploaded rebills")

    rebills.initial_payment_date =
rebills.initial_payment_date.dt.tz_localize(None)

    # divide users into cohorts of cohorts_duration days
    rebills['diff_from_last'] = (rebills.initial_payment_date -

rebills.initial_payment_date.min()).dt.days // cohorts_duration

    rebills['max_cohort_initial_payment_date'] = rebills.groupby(
        params +
['diff_from_last']).initial_payment_date.transform('max')
```

```
    rebills['max_cohort_assumption_rebill_date'] =
rebills.max_cohort_initial_payment_date + \
                                                  pd.to_timedelta(
rebills.rebill_duration * (rebills.rebill_number - 1), unit='d')

    rebills.sort_values(params + ['rebill_number'], ascending=False,
inplace=True)

    coeffs_rebills = pd.DataFrame()

    for iterator, weight in zip(
            range(len(params), 0, -1),
            range(0, -len(params), -1)
    ):
        parameters = params[:iterator]
        parameters_not_included = params[iterator:]

        rebills_data = rebills.groupby(parameters + ['diff_from_last',
'rebill_number',
'max_cohort_assumption_rebill_date',
'rebills_amount_per_period', 'rebill_duration',
'trial_duration']).payments.sum().reset_index()

        rebills_data['total_payments'] =
rebills_data.groupby(parameters +
['diff_from_last']).payments.transform('max')

        rebills_data = rebills_data[rebills_data.total_payments >=
total_payments_threshold]

        for i in rebills_data[rebills_data.max_rebill_number >
2][parameters].drop_duplicates().values:
            rebills_fit_data =
rebills_data[(np.all(rebills_data[parameters] == i, axis=1)) &
(rebills_data.max_rebill_number > 2)]

            diff_from_last_for_forecast =
rebills_fit_data.diff_from_last.sort_values().unique()[-20:]
            rebills_fit_data =
rebills_fit_data[rebills_fit_data.diff_from_last.isin(diff_from_last_f
or_forecast)]

            fit_matrix = pd.pivot_table(rebills_fit_data,
                                        index='diff_from_last',
columns='rebill_number',
```

```
values='payments').reset_index(drop=True).values

            fit_matrix = handle_fit_matrix(fit_matrix)

            sbg_model = sbg()
            sbg_model.fit(fit_matrix)

            if not sbg_model.success_flag:
                continue

            max_rebill_number_for_group =
rebills_fit_data.rebills_amount_per_period.values[0]

            coeffs_rebills_inner = pd.DataFrame(np.array([i]).tolist()
* max_rebill_number_for_group,
                                                columns=parameters)

            predict =
sbg_model.predicted_survival(max_rebill_number_for_group - 1)
            coeffs_rebills_inner['rebill_rate'] = predict

            for i in parameters_not_included:
                coeffs_rebills_inner[i] =
[rebills[i].unique().tolist()] * coeffs_rebills_inner.shape[0]

                coeffs_rebills_inner = coeffs_rebills_inner.explode(i)
            coeffs_rebills_inner['weight'] = weight

            coeffs_rebills = pd.concat([coeffs_rebills,
coeffs_rebills_inner], ignore_index=True)

    # choose the coeffs with the largest weights
    coeffs_rebills =
coeffs_rebills.sort_values('weight').drop_duplicates(params +
['rebill_number'],

keep='last').reset_index(drop=True)

    coeffs_rebills = coeffs_rebills.sort_values(params +
['rebill_number'],

ascending=False).reset_index(drop=True)
    coeffs_rebills['rebill_rate'] =
coeffs_rebills.groupby(params).rebill_rate.cumsum()

    coeffs_rebills = coeffs_rebills[params + ['rebill_number',
'rebill_rate', 'weight']]

    insert_dataframe_into_bq_table(coeffs_rebills,
                                   'diploma-417512.final.' +
table_to_which_insert, append=False)
```

ДОДАТОК Г Візуалізація даних в Tableau





10/27/2021 2/13/2022

Country
(All)

App Name
(All)

Media Source
(All)

Cohorts by Month

Month of Acquisition Date	Cost	Actual Revenue	CPA	Purchases	Payments	Purchases -> Payments	Cancel Rate 1d	Cancel Rate 7d	Cancel Rate 30d	Refund Rate	Actual ARPS
Grand Total	$3,571,876.7	$740,718.6	$327.7	48,176	22,638	228.9%	12.8%	33.6%	185.82%	38.76%	$77.4
February 2022	$409,755.8	$56,453.0	$72.3	5,671	2,423	42.7%	3.0%	7.9%	46.74%	6.63%	$10.0
January 2022	$1,918,638.3	$337,739.1	$85.7	22,396	11,030	49.2%	1.7%	4.5%	43.98%	10.78%	$15.1
December 2021	$809,367.1	$188,987.7	$74.9	10,807	5,076	47.0%	2.4%	5.3%	36.72%	8.59%	$17.5
November 2021	$362,266.0	$130,735.9	$46.3	7,823	3,427	43.8%	2.9%	6.0%	28.80%	6.14%	$16.7
October 2021	$71,629.5	$26,822.8	$48.6	1,479	682	46.1%	2.9%	6.0%	27.59%	6.63%	$18.1

Додаток Д СТРУКТУРА ДАНИХ BigQuery

Схема таблиці «insights»

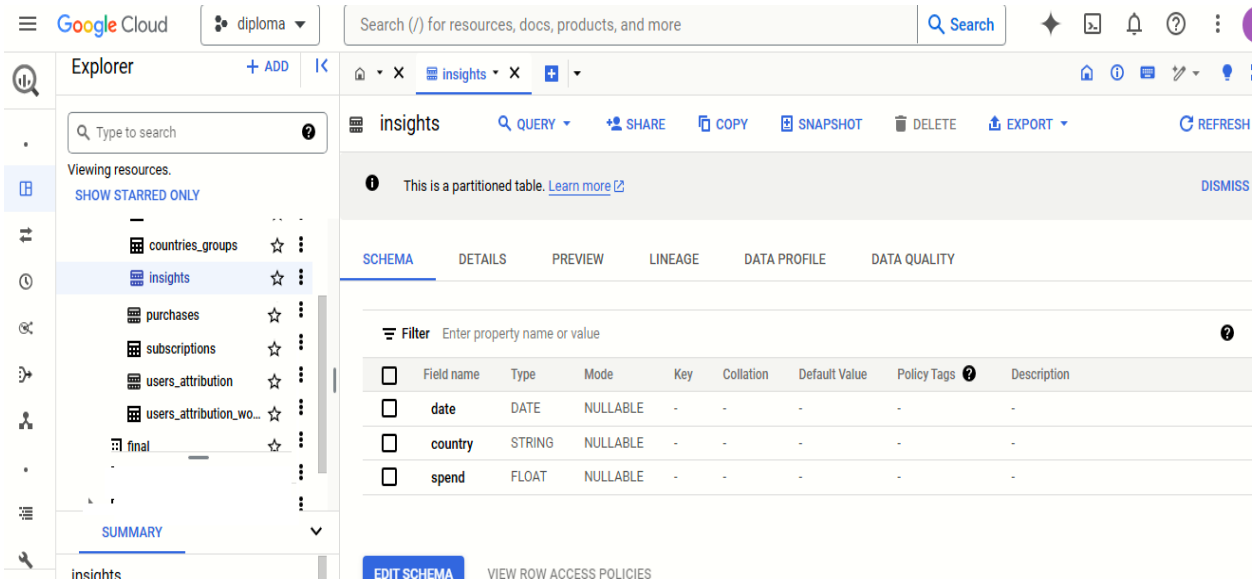


Схема таблиці «Purchases»

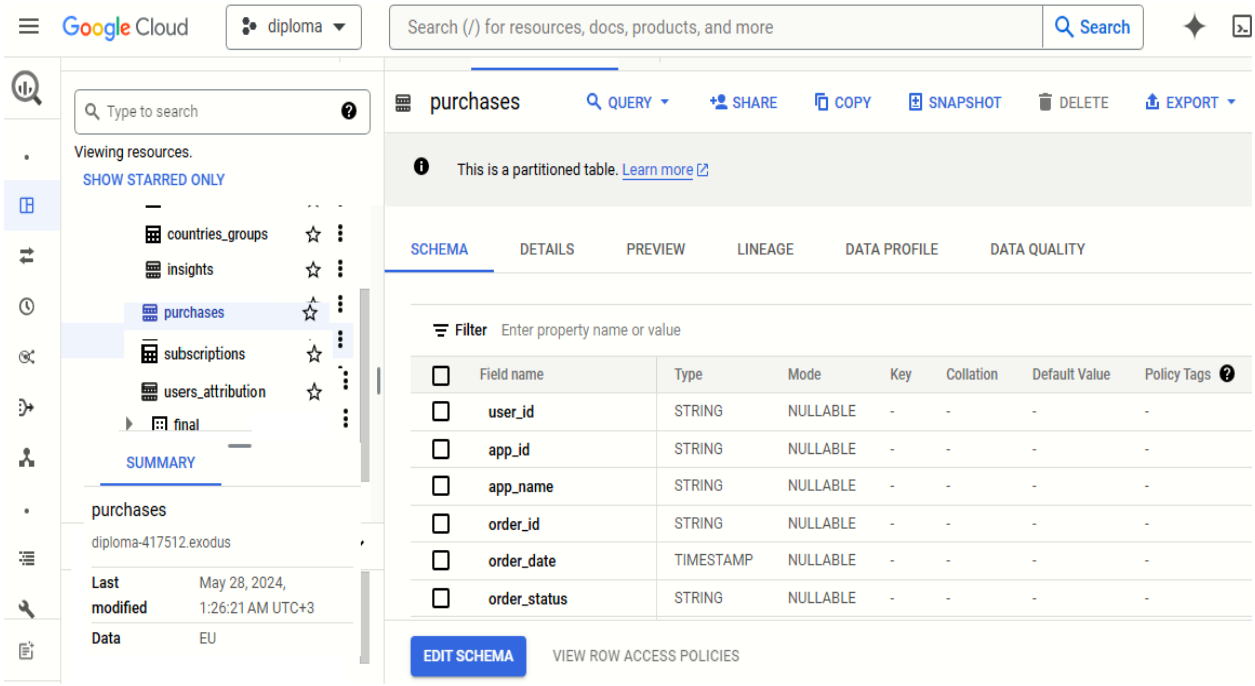


Схема таблиці «Users_attrribution»

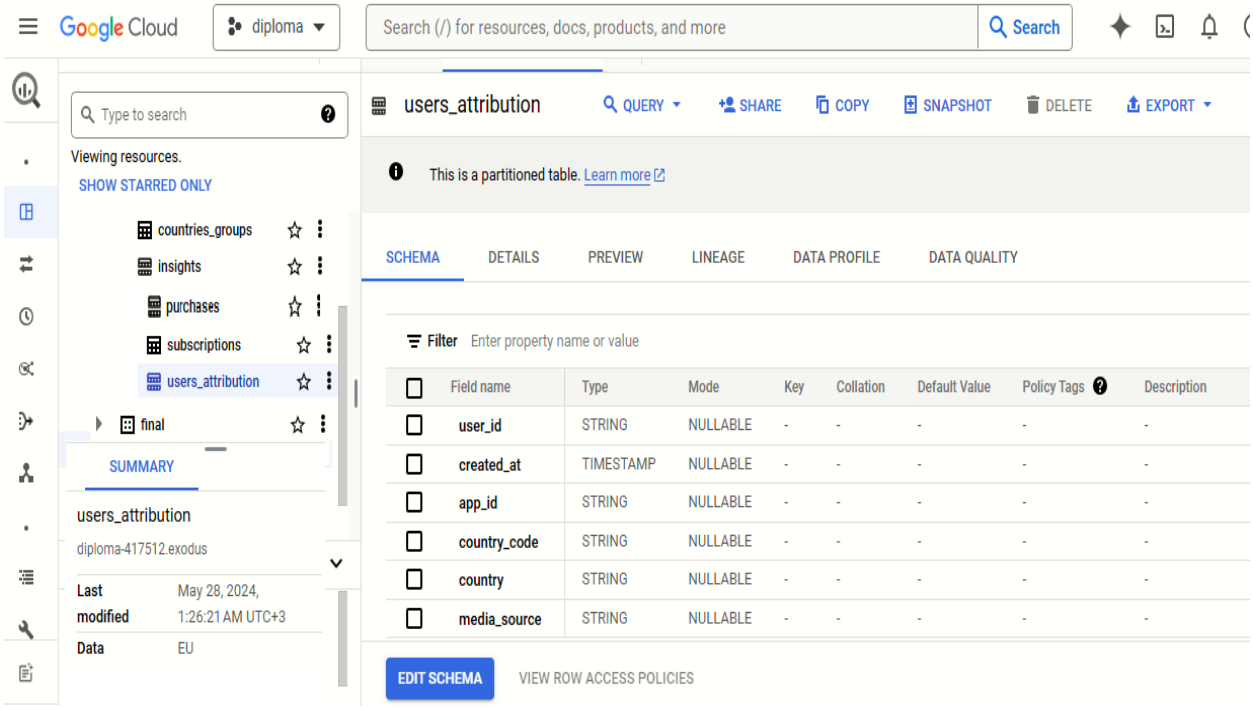


Схема таблиці «Subscribtions»

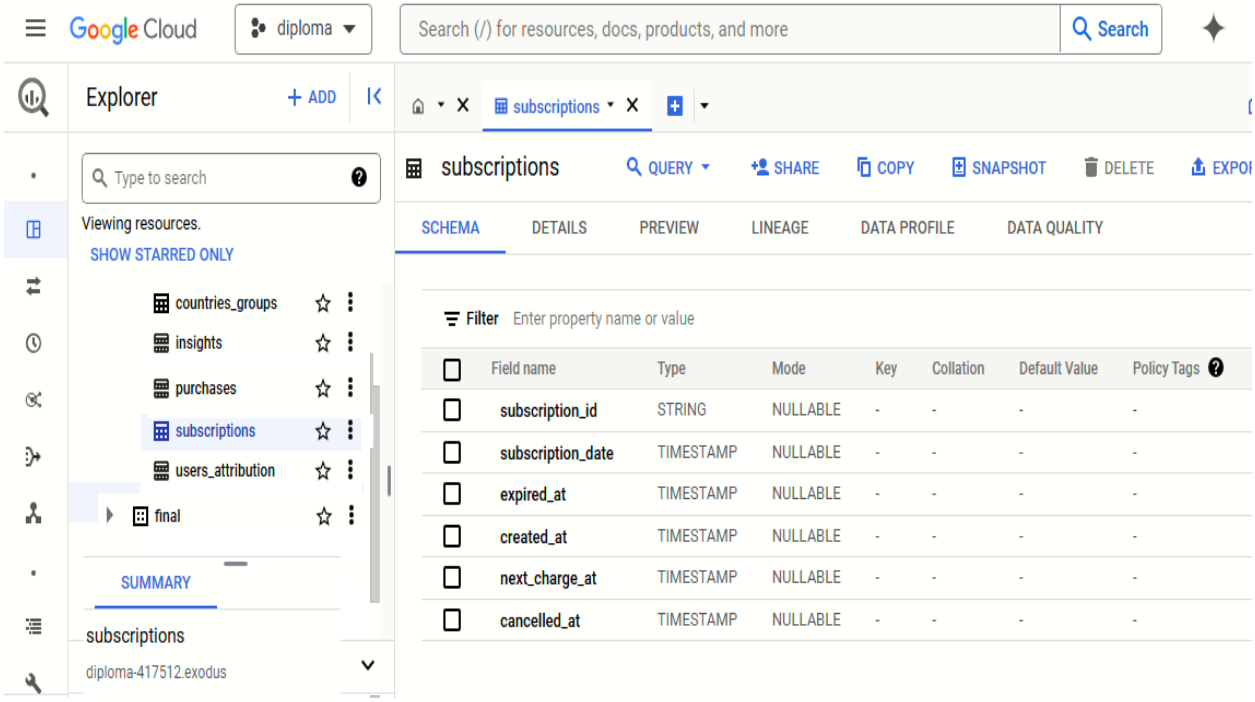


Схема таблиці «Countries_groups»

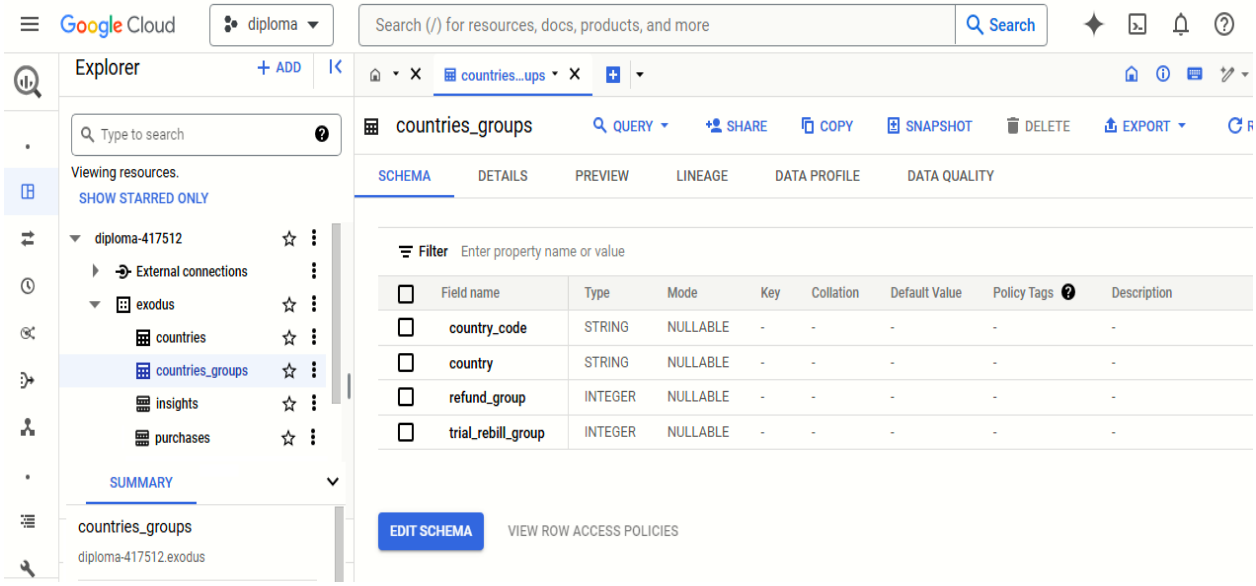


Схема таблиці «Actual_Revenue»

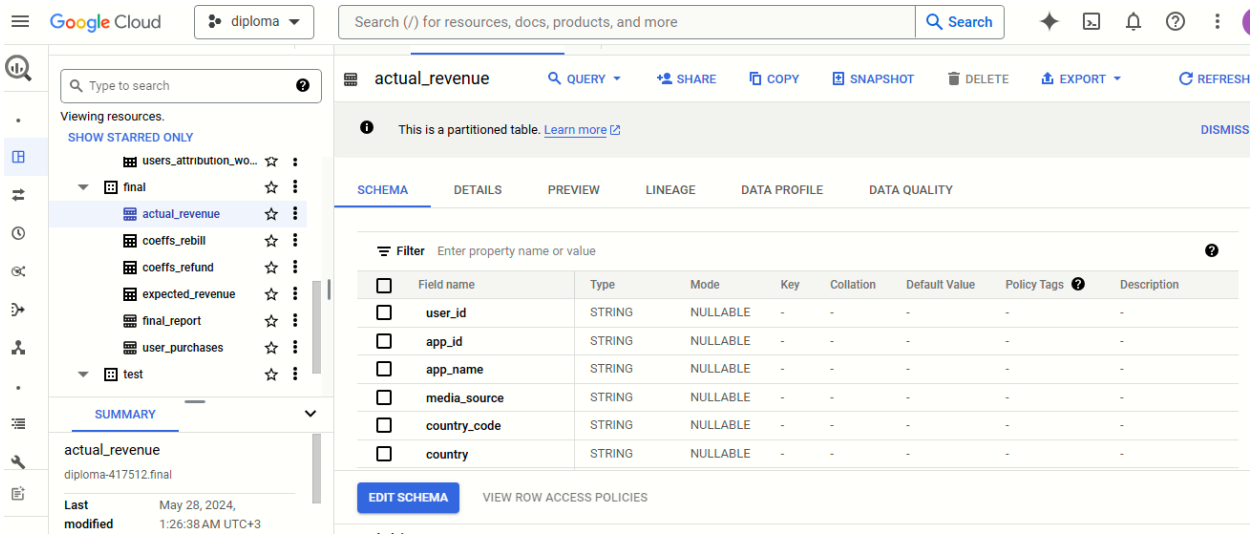


Схема таблиці «User_Purchases»

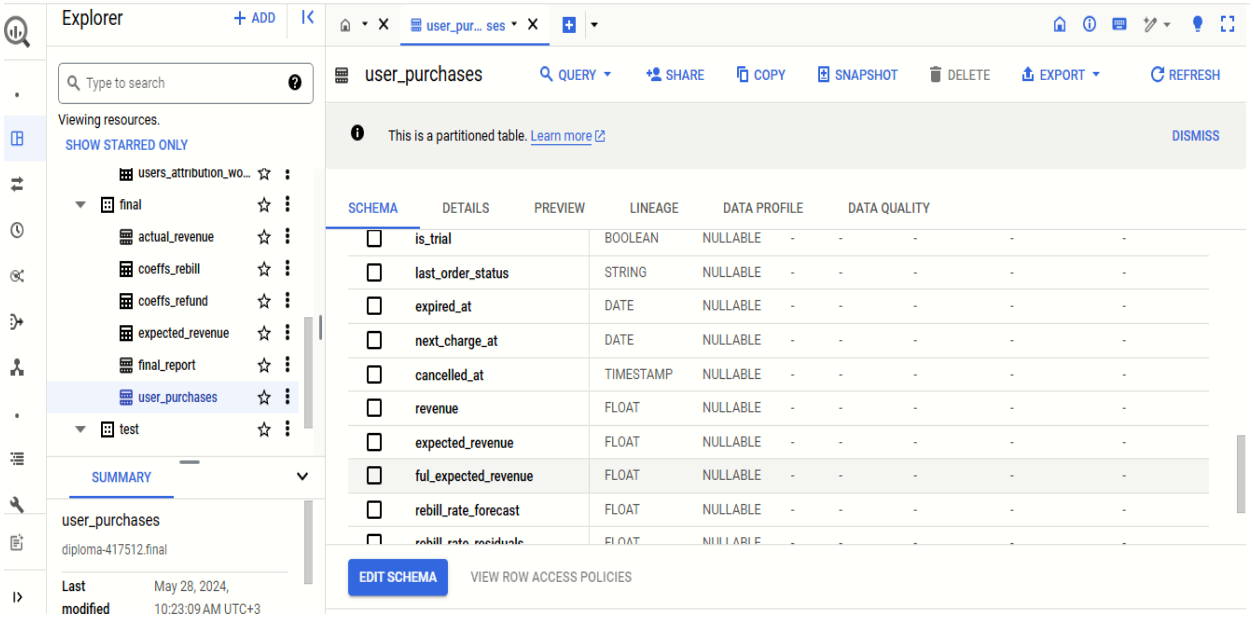


Схема таблиці «Coeffs_refund»

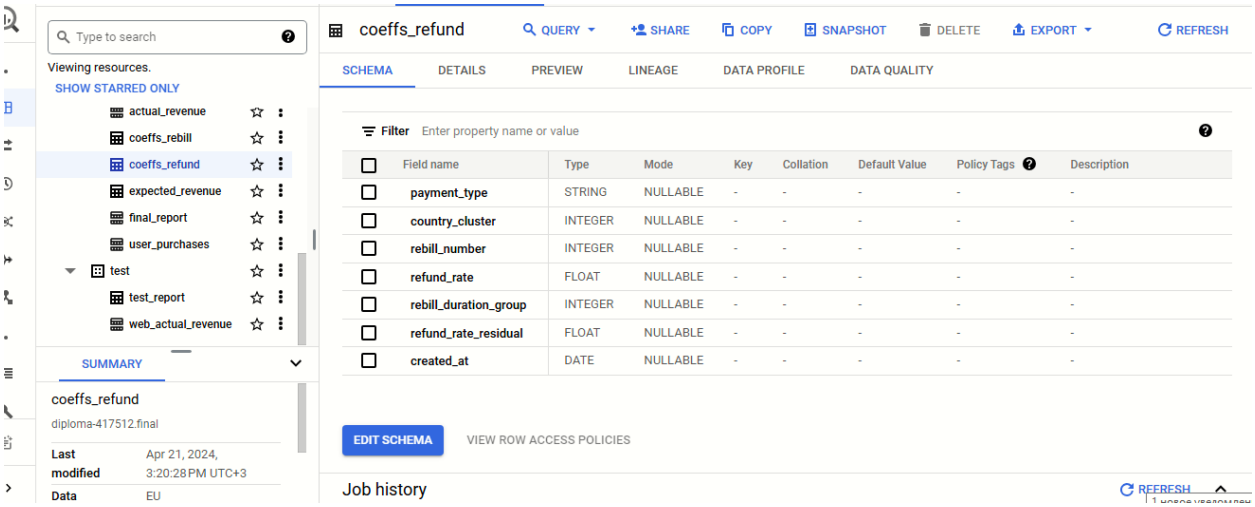


Схема таблиці «Expected_revenue»

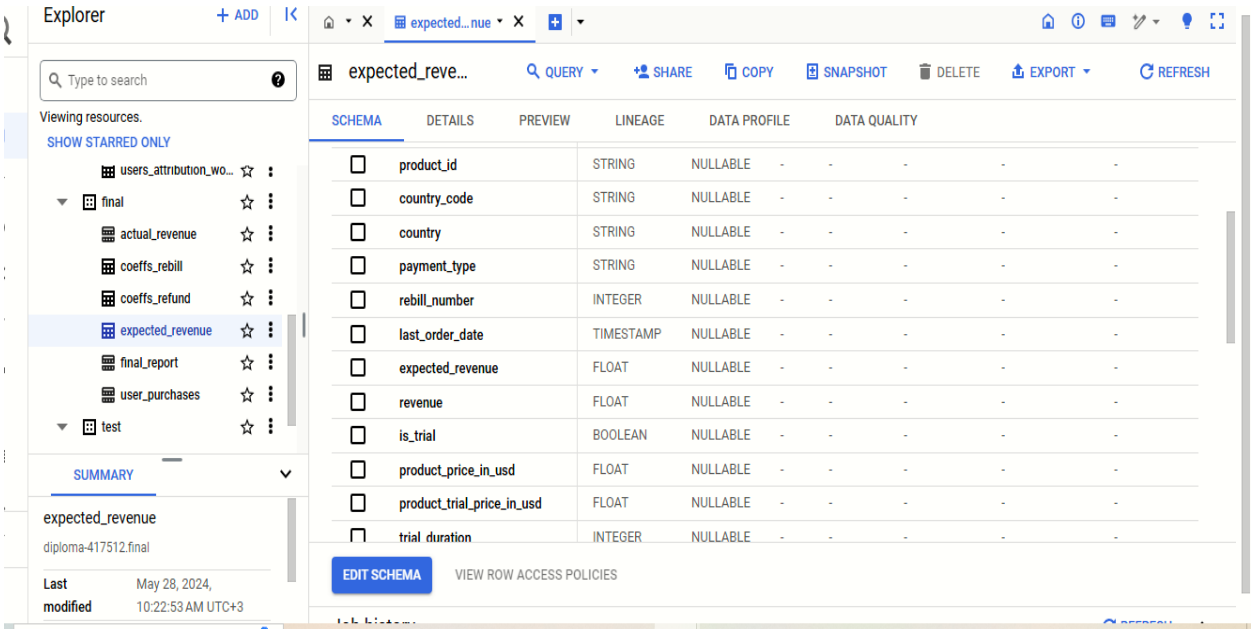


Схема таблиці «Final_Report»