Phong Hoang
Nectar Data Assessment Report
Jan 2, 2025

Nectar Data Assessment Report

1. Tangible Deliverable

Question 1: **Extract user data** - Write a SQL query to extract the id, created_at, and last_active from the users table for users who signed up in the last 30 days.

```
SELECT id, created_at, last_active
FROM users
WHERE created_at >= DATE_SUB(CURRENT_DATE, INTERVAL 30 DAY);
```

Explanation:

1. Filter Recent Signups
    - DATE_SUB(CURRENT_DATE, INTERVAL 30 DAY). Calculate the date exactly 30 days before today. Ensures only users who signed up on or after this date are included, filtering out older users.

Note: If the dates in the dataset differ from the actual "current date" (for example, it's static test data), we can adjust the filter accordingly using a specific date instead of CURRENT_DATE.

Question 2: **Integrate payment stream data** - Write a SQL query or Python script to join the user data from the users table with the payment transaction stream. Calculate the total revenue generated by users who signed up in the last 30 days and made a payment during that time.

```
SELECT SUM(p.amount) AS total_revenue

FROM users u

JOIN payments p

ON u.id = p.user_id

WHERE u.created_at >= DATE_SUB(CURRENT_DATE, INTERVAL 30 DAY)
```

AND p.payment_date >= u.created_at;

Explanation:

1. Join Tables:
   - The users table is joined with the payments table using the user_id column, matching users with their payment transactions.
2. Filter New Signups:
   - u.created_at >= DATE_SUB(CURRENT_DATE, INTERVAL 30 DAY). Filters users who signed up in the last 30 days.
3. Filter Payments After Signup:
   - p.payment_date >= u.created_at. Ensures only payments made after the user signed up are considered.
4. Aggregate Revenue:
   - SUM(p.amount) calculates the total revenue from all qualifying payments.

Question 3: **Track message activity** - Write a SQL query to identify active users based on messages. Specifically, count distinct users who have sent at least one message in the last 30 days.

SELECT COUNT(DISTINCT c.user_id) AS active_users

FROM messages m

JOIN conversations c

ON m.conversation_id = c.conversation_id

WHERE m.sent_at >= DATE_SUB(CURRENT_DATE, INTERVAL 30 DAY);

Explanation:

1. Join Tables
   - ON m.conversation_id = c.conversation_id. Combines the messages and conversations tables to link messages with the users who sent them.
2. Filter Recent Messages:
   - m.sent_at >= DATE_SUB(CURRENT_DATE, INTERVAL 30 DAY). Ensures only messages sent in the last 30 days are considered.

3. Count Unique Users:
    - COUNT(DISTINCT c.user_id). Counts the number of distinct users (user_id) who sent messages within the filtered period.

Question 4: **Calculate the 30-day conversion rate**: i.e., what percentage of users who signed up in the last 30 days made at least one payment or sent a message. Assume last_active does not cover this case

```
WITH recent_users AS (

   SELECT id AS signup_user_id

   FROM users

   WHERE created_at >= DATE_SUB(CURRENT_DATE, INTERVAL 30 DAY)),

engaged_users AS (

   SELECT DISTINCT payment_user_id

   FROM payments

   WHERE payment_date >= DATE_SUB(CURRENT_DATE, INTERVAL 30 DAY)

   UNION

   SELECT DISTINCT message_user_id

   FROM messages

   JOIN conversations ON messages.conversation_id = conversations.conversation_id

   WHERE messages.sent_at >= DATE_SUB(CURRENT_DATE, INTERVAL 30 DAY))

SELECT

   COUNT(DISTINCT engaged_users.payment_user_id) * 100.0 / COUNT(DISTINCT
recent_users.signup_user_id) AS conversion_rate

FROM recent_users
```

LEFT JOIN engaged_users

ON recent_users.signup_user_id = engaged_users.payment_user_id;

Explanation:

The query's purpose is to calculate the 30-day conversion rate, which is the percentage of users that joined up within the last 30 days and have actively engaged with the platform. To accomplish this, the first step is to identify the users who signed up during the last 30 days. These users serve as the base population for conversion calculations. By filtering the users table by signup date (created_at), we may isolate this subset of recent users. This guarantees that we only focus on the users who are relevant to the analysis. Next, we identify the users who actively used the platform within the same period. Engagement is calculated by querying two tables: the payments table for users who made payments in the last 30 days, and the messages table for users who sent messages during the same period. Because some users may engage in both activities, we combine these two groups of people with a UNION to eliminate double-counting and create a distinct list of engaged users. Once we have both sets of users—recent signups and engaged users—we compare them to see which recently signed-up users interacted with the platform. This comparison allows us to determine the conversion rate, which is equal to the number of engaged users divided by the total number of recent signups multiplied by 100. The result is a percentage that quantifies how many new users transitioned into active engagement.

Question 5: **Calculate the 30-day retention rate:** percentage of users who continue to engage with the product 30 days after their initial sign-up or purchase Assume last_active does not cover this case

WITH initial_interaction AS (

 SELECT u.id AS user_id, LEAST(u.created_at, MIN(p.payment_date)) AS initial_date

   FROM users u

   LEFT JOIN payments p ON u.id = p.user_id

   GROUP BY u.id),

engaged_users AS (

```
    SELECT i.user_id
    FROM initial_interaction i
    JOIN payments p ON i.user_id = p.user_id
    WHERE p.payment_date > i.initial_date
     AND p.payment_date <= DATE_ADD(i.initial_date, INTERVAL 30 DAY))
SELECT
    COUNT(DISTINCT e.user_id) * 100.0 / COUNT(DISTINCT i.user_id) AS retention_rate
FROM initial_interaction i
LEFT JOIN engaged_users e ON i.user_id = e.user_id;
```

First, the query identifies the initial interaction date for each user. This is either the signup date (created_at) from the users table or the date of their first payment (payment_date) from the payments table, whichever is earlier. This ensures we have a baseline for measuring user activity. Next, the query identifies users who engaged again within 30 days of their initial interaction. This is done by filtering payments to include only those made by users after their initial interaction date and within the subsequent 30 days. This step ensures we capture users who demonstrated continued engagement during the defined retention period. Finally, the retention rate is calculated as the ratio of users who re-engaged to the total number of users. Specifically, the numerator is the count of distinct users who made a payment within 30 days, and the denominator is the total number of users. This ratio is then multiplied by 100 to express the retention rate as a percentage.

2. **Open Ended Deliverables**

If you are to run an A/B test, and the feature is to be rolled out to 50% of new users / 50% control, how would you evaluate its effect on 30-day total conversion rate and 30-day retention rate?

Here is the step I would take in this situation

1. Define the Experiment Setup

To start, I define the experiment's objective: evaluating how the new feature impacts the 30-day conversion rate and 30-day retention rate. I divide new users into two groups:

- Control Group: Users who do not experience the new feature.
- Test Group: Users exposed to the new feature.

I ensure the split is random and balanced, assigning 50% of new users to each group using a deterministic method, like hashing user IDs. This ensures consistency if I need to reprocess the data. The metrics to focus on are:

- Conversion Rate: Percentage of users performing the target action Retention Rate: Percentage of users still active 30 days post-signup.

This setup ensures fairness and removes bias from group assignment. By clearly defining the metrics upfront, I'm confident that the results will be relevant and actionable.

2. Data Preparation

Collect and Load Data: I ensure access to all the necessary datasets, including:

- Users Data: Contains user-level information such as user_id, created_at (signup date), and last_active (most recent activity date).
- Payments Data: Tracks user transactions, including user_id, payment_date, and amount.
- Engagement Data: Records user activities, such as message_sent or sessions, with timestamps (sent_at).
- Experimental Metadata: Includes group assignments (control vs. test) if pre-existing or criteria to assign them.

Data cleaning is crucial here. To ensure time-based analysis works smoothly, I convert all relevant date columns to datetime format and filter the dataset for new users within the last 30 days. I handle missing values, remove invalid timestamps, and confirm the integrity of the data. For instance, I check for duplicate user IDs and confirm timestamps align with expected ranges. My goal is to have a clean, ready-to-analyze dataset with obvious group allocations.

By the end of this process, I'll have two clearly defined groups with no data inconsistencies, laying the groundwork for reliable analysis.

3. Analyzing Metrics

In addition to conversion rate and retention rate, I believe that collecting supplementary metrics can provide a more comprehensive view of the A/B test.

3.1. Engagement Metrics: Understanding user interactions with the feature is crucial for measuring its adoption, assessing its impact on engagement beyond retention or conversion, and determining whether it is being used as intended.

Daily Active Users (DAU):

- Number of unique users actively using the app on a daily basis.
- Helps measure user stickiness and engagement frequency.

Session Count:

- Average number of sessions per user within the first 30 days.
- Reveals whether the feature encourages more frequent usage.

Feature-Specific Interactions:

- Count of how often users engage directly with the new feature.

3.2. Revenue Metrics: Beyond conversion rates, it’s essential to analyze how much revenue the feature generates and its impact on spending behavior, such as higher revenue per user or faster purchases. This ensures the feature improves revenue quality and identifies potential unintended consequences, like reduced spending per transaction.

Average Revenue Per User (ARPU):

- Total revenue divided by the total number of users in each group.
- Indicates the overall financial impact of the feature.

Revenue per Paying User (ARPPU):

- Revenue generated only from users who made a payment.

- Helps assess whether the feature increases spending among converters.

Time to First Purchase:

- Average time it takes for users to make their first payment.
- A shorter time indicates that the feature drives faster conversions.

3.3 Funnel Metrics: Break down the user journey to identify where the feature has the most significant impact for each groups

Step-by-Step Conversion Rates:

For example:

Signup → First Interaction

First Interaction → Conversion

Conversion → Retention

=> Helps identify where users drop off in the process.

Feature Adoption Rate:

- Percentage of users in the test group who engage with the feature at least once.
- Indicates how accessible and appealing the feature is to users.

3.4. User Experience Metrics: xAssessing user experience ensures the feature meets user expectations while identifying and addressing potential usability issues or friction that could hurt adoption, satisfaction, or overall engagement.

Error Rate:

- Percentage of users encountering errors while using the feature.

Completion Rate:

- Percentage of users successfully completing actions related to the feature.

Time on Feature:

- Average time users spend interacting with the feature.
- A short time could indicate ease of use, while an excessively long time might signal confusion or friction.

3.5. Behavioral Segmentation Metrics: Segmenting users helps identify whether the feature works better for certain types of users, uncovering uneven effects across groups and highlighting opportunities for tailoring or optimization.

Retention and Conversion by Segment:

- Example segments:
    - Device type (mobile vs. desktop)
    - Geographic region
    - Signup source (organic vs. paid)
- Identifies whether the feature is more effective for specific user groups.

Churn Rate:

- Percentage of users who leave the platform after interacting with the feature.
- Helps measure unintended negative consequences.

3.6. Long-Term Metrics: If we can track users beyond the 30-day window, longer-term metrics can reveal lasting effects.

90-Day Retention Rate:

- Measures whether the feature drives sustained engagement.

Lifetime Value (LTV):

- Revenue generated by users over their entire lifecycle.

Churn Post-Adoption:

- Percentage of users who churn after adopting the feature.

4. Statistical Testing

To ensure the observed differences aren't due to random chance, I perform statistical tests. The **chi-square test** is ideal for evaluating conversion and retention rates in A/B testing as it compares proportions between groups and determines if differences are statistically significant. It handles categorical data well, works effectively with large sample sizes, and does not rely on assumptions about data distribution. By providing a clear p-value, it quantifies whether observed differences are due to chance, making it a robust and reliable choice for A/B test analysis.

5. Visualization and Reports

The final step of the A/B testing process is where I collect a report that blends visualizations with a detailed examination of the results to come to conclusive and actionable recommendations.

Visualization is the main means through which we can present the findings effectively. When it comes to conversion and retention rates, I use parallel bar charts to compare the test ones with those of the control group singling out all the differences by means of clear images. Confidence intervals or error bars are used to visualize the distributions of the metrics and reveal the degree of statistical uncertainty. In connection with the outcomes, I developed grouped bar charts that depict the average revenue per user and the revenue per paying user. This new and advanced approach allows a more detailed and comprehensive look at how an introductory feature is related to spending behavior. A funnel chart is a good example of the step-by-step process from signups to conversions and retention, thus helping one identify the exact stage where the user journey has the highest dropout rate. These illustrations not only make the results clearer to all the partners but also make it possible to intuitively understand the effect of the test more effectively.

The report is opened with a short overview that outlines the objectives of the A/B test, the criteria applied, and the group setup. The part of the results provides the exact outcomes for conversion rates, retention rates, and revenue metrics. For each metric, I present the figures for the test as well as the control groups. Then, I analyze whether the differences between the measures are statistically significant or not and illustrate the numerical correlation using the respective p-values. Also, I describe any observed trends or exceptions, e.g., whether the feature led to a better conversion rate at the expense of decreased retention or decreased spending quality. At the end, I provide the overall effects of the test. To clarify, if the functionality has led to increased conversion but has not been as effective in retention, I would underscore the operation and balance aspects. Context-specific views are also expressed, such as the success of the sector in the case record or the effect of such a trend.

Finally, the recommendations lay out the next actions. If the feature worked well, I recommend implementing it extensively, possibly focusing on segments where it did best. If the results were inconclusive, I would propose iterating on the feature and conducting additional testing. For a negative outcome, I advise avoiding a full rollout while identifying chances for refinement via subgroup analysis. This technique guarantees that stakeholders receive a clear and thorough overview of the A/B test, allowing them to make confident data-driven decisions. Please let me know if you require any additional refining!