Case Studay 1 Report
Chuan (Jasper) Liu
2025-04-07

Case Study 1: How does a bike-share navigate speedy success?

In this case study, we will perform data analysis for a fictional bike-share company in order to help them come up with better marketing plans that will effectively bring more platform users. The company provided their 2-year shared bike usage data for both their members and non-member users.

Ask：Questions to be addressed

1. How do annual members and casual riders use Cyclistic bikes differently?
2. Why would casual riders buy Cyclistic annual memberships?
3. How can Cyclistic use digital media to influence casual riders to become members?

Prepare

1. Access data: this is public data from the website:
https://divvytripdata.s3.amazonaws.com/index.html, given by the class, we will take previous 6 months of cyclistic data.

2. The data consists of 6 csv files: The files for the months of March, April, November, and December in 2024 and January and February in 2025.4.

3. The data is first-party data collected by Cyclist so there is low chance of bias, but due to it being the companies own data the credibility is very high.

4. The data is open source and covered by licence : https://divvybikes.com/data-license-agreement

Process

```
-- Below is my analyze for Google Data Analytics Capstone Case Study 1.
-- Step 1, upload all data (There are a few data files cannot be uploaded because
over size limit, so I will use partial data to present my analysis) and combine
them together.
CREATE TABLE `my-first-project-452000.2024_divvy_ridedata.2024_total` AS
FROM
(SELECT * FROM `my-first-project-452000.2024_divvy_ridedata.202403`
UNION ALL
SELECT * FROM `my-first-project-452000.2024_divvy_ridedata.202404`
UNION ALL
SELECT * FROM `my-first-project-452000.2024_divvy_ridedata.202411`
UNION ALL
SELECT * FROM `my-first-project-452000.2024_divvy_ridedata.202412`
UNION ALL
SELECT * FROM`my-first-project-452000.2024_divvy_ridedata.202501`
UNION ALL
SELECT * FROM `my-first-project-452000.2024_divvy_ridedata.202502`)
--Check the total rows in the new table
select count(*)
from `my-first-project-452000.2024_divvy_ridedata.2024_total`
```

（feedbcak:1520728 rows）

```sql
--Check ride_id length
SELECT LENGTH(ride_id) AS length_ride_id, COUNT(ride_id) AS no_of_rows
FROM `my-first-project-452000.2024_divvy_ridedata.2024_total`
GROUP BY length_ride_id
```

length_ride_id ▼	no_of_rows ▼
16	1520728

```sql
--Check bike type
SELECT DISTINCT rideable_type, COUNT(rideable_type) AS no_of_trips
FROM `my-first-project-452000.2024_divvy_ridedata.2024_total`
GROUP BY rideable_type
```

rideable_type ▼	no_of_trips ▼
electric_bike	862715
classic_bike	658013

Analyze

SEPT 1: Clean

```sql
--Create a cleaned table with all values we need for analysis.
create table `my-first-project-452000.2024_divvy_ridedata.2024_tota_cleaned_2` AS (
  select a.ride_id, rideable_type, started_at, ended_at,
    ride_length,
--split started data form intoDAY/MONTH
    CASE EXTRACT(DAYOFWEEK FROM started_at)
      WHEN 1 THEN 'SUN'
      WHEN 2 THEN 'MON'
      WHEN 3 THEN 'TUES'
      WHEN 4 THEN 'WED'
      WHEN 5 THEN 'THURS'
      WHEN 6 THEN 'FRI'
      WHEN 7 THEN 'SAT'
    END AS day_of_week,
    CASE EXTRACT(MONTH FROM started_at)
      WHEN 1 THEN 'JAN'
      WHEN 2 THEN 'FEB'
      WHEN 3 THEN 'MAR'
      WHEN 4 THEN 'APR'
      WHEN 5 THEN 'MAY'
      WHEN 6 THEN 'JUN'
      WHEN 7 THEN 'JUL'
      WHEN 8 THEN 'AUG'
      WHEN 9 THEN 'SEP'
      WHEN 10 THEN 'OCT'
      WHEN 11 THEN 'NOV'
      WHEN 12 THEN 'DEC'
    END AS month,
    start_station_name, end_station_name,
    start_lat, start_lng, end_lat, end_lng, member_casual
--mark our old file as "a",and join two files with a now column "ride_length"
  FROM `my-first-project-452000.2024_divvy_ridedata.2024_total` a
  JOIN (
    SELECT ride_id, TIMESTAMP_DIFF(ended_at, started_at, minute) AS ride_length
```

```
      FROM `my-first-project-452000.2024_divvy_ridedata.2024_total`) b
   ON a.ride_id = b.ride_id
--remove all trash rows
   WHERE
      start_station_name IS NOT NULL AND
      end_station_name IS NOT NULL AND
      end_lat IS NOT NULL AND
      end_lng IS NOT NULL AND
      ride_length > 1 AND ride_length < 1440)
```

STEP 2:Analyze

```
-- calculate max,min,mean value (in minutes)
select max(ride_length) as max, min(ride_length) as min, avg(ride_length) as mean
from `my-first-project-452000.2024_divvy_ridedata.2024_tota_cleaned_2`
```
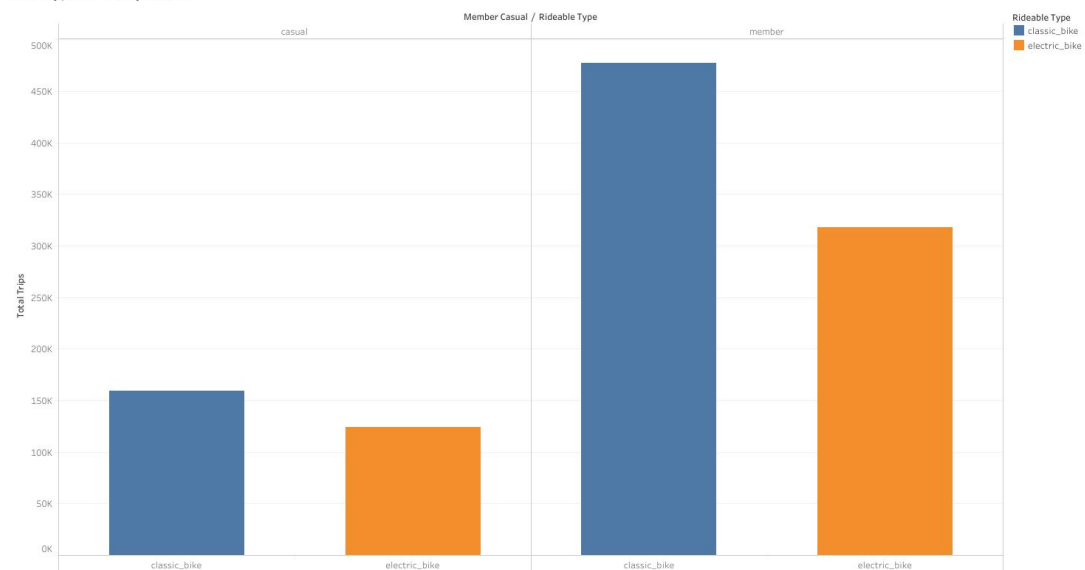
max ▾	min ▾	mean ▾
1437	2	13.33036354146...

```
-- bikes types used by riders
SELECT member_casual, rideable_type, COUNT(*) AS total_trips
FROM `my-first-project-452000.2024_divvy_ridedata.2024_tota_cleaned_2`
GROUP BY member_casual, rideable_type
ORDER BY member_casual, total_trips
```
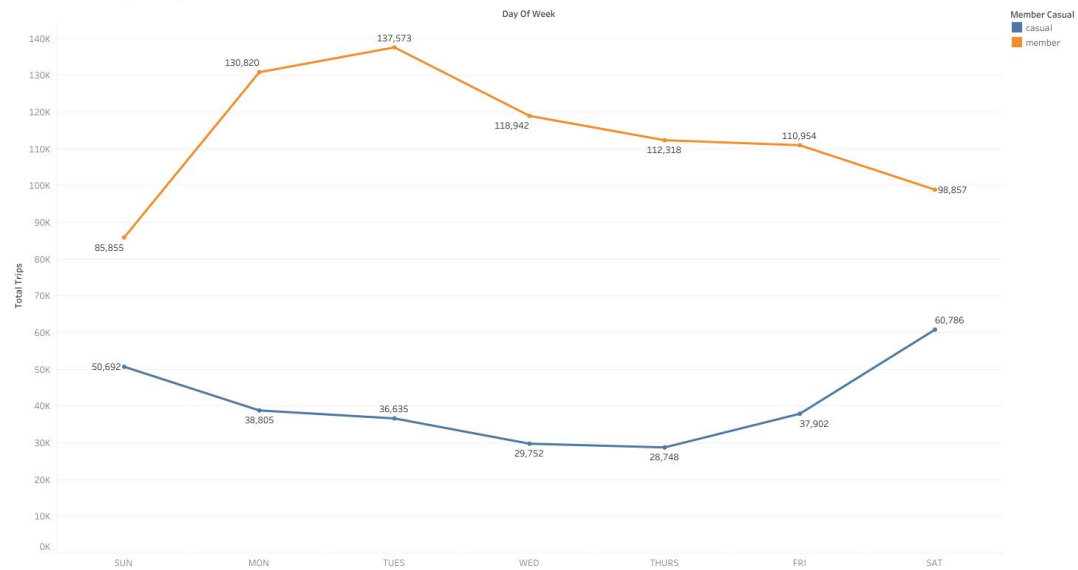
Bike Types Used by Riders

```
-- compare number of trips per month between member_casual (Since the insufficient
data, we cannot generate a meaningful conclusion group by month)
SELECT month, member_casual, COUNT(ride_id) AS total_trips
FROM `my-first-project-452000.2024_divvy_ridedata.2024_tota_cleaned_2`
GROUP BY month, member_casual
ORDER BY month


-- compare number of trips per day of week between member_casual
SELECT day_of_week, member_casual, COUNT(ride_id) AS total_trips
FROM `my-first-project-452000.2024_divvy_ridedata.2024_tota_cleaned_2`
GROUP BY day_of_week, member_casual
ORDER BY day_of_week, member_casual
```
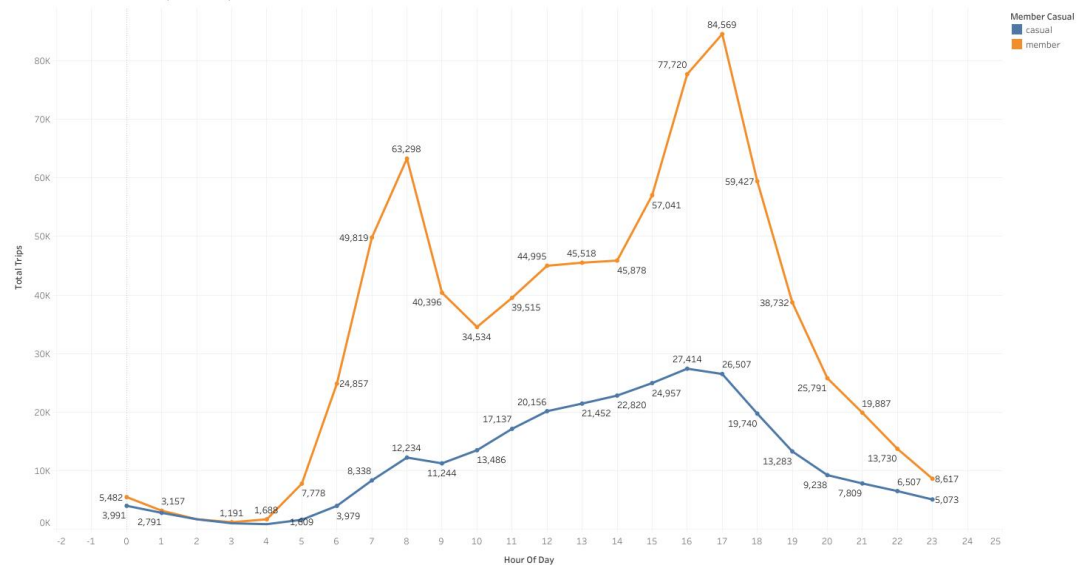
Number of Trips (pre day)


```
-- compare number of trips per hour between mumber_casual
SELECT EXTRACT(HOUR FROM started_at) AS hour_of_day, member_casual, COUNT(ride_id)
AS total_trips
FROM `my-first-project-452000.2024_divvy_ridedata.2024_tota_cleaned_2`
GROUP BY hour_of_day, member_casual
ORDER BY hour_of_day, member_casual
```

Total Number of Trips (pre hour)

```sql
-- compare average ride_length per month between member_casual(Since the
insufficient data, we cannot generate a meaningful conclusion group by month)
SELECT month, member_casual, AVG(ride_length) AS avg_ride_duration
FROM `my-first-project-452000.2024_divvy_ridedata.2024_tota_cleaned_2`
GROUP BY member_casual, month
ORDER BY month

-- compare average ride_length per day of week between member_casual
SELECT day_of_week, member_casual, AVG(ride_length) AS avg_ride_duration
FROM `my-first-project-452000.2024_divvy_ridedata.2024_tota_cleaned_2`
GROUP BY day_of_week, member_casual
ORDER BY day_of_week
```
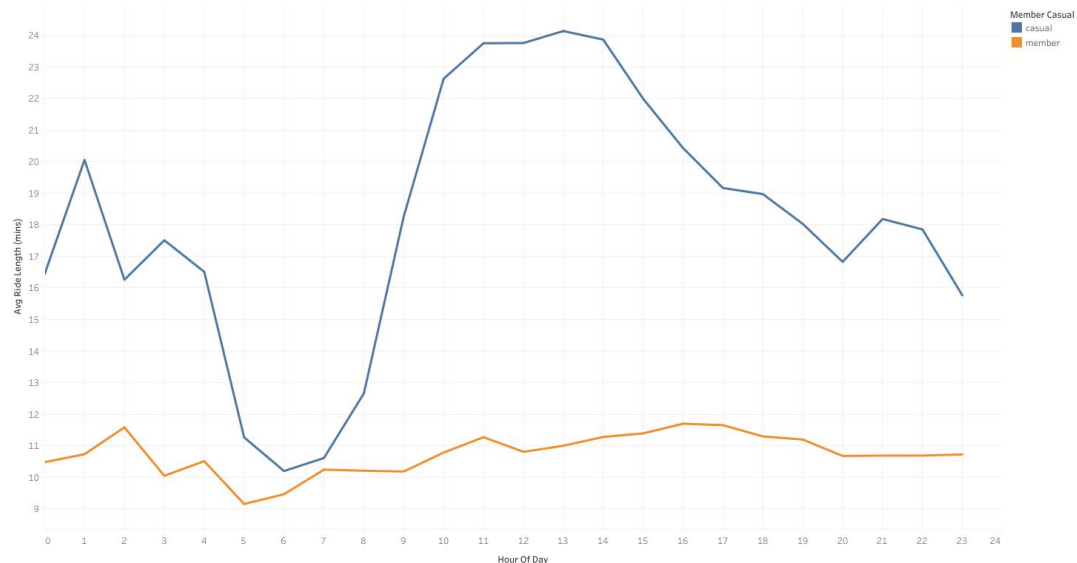
Average Ride Length per day
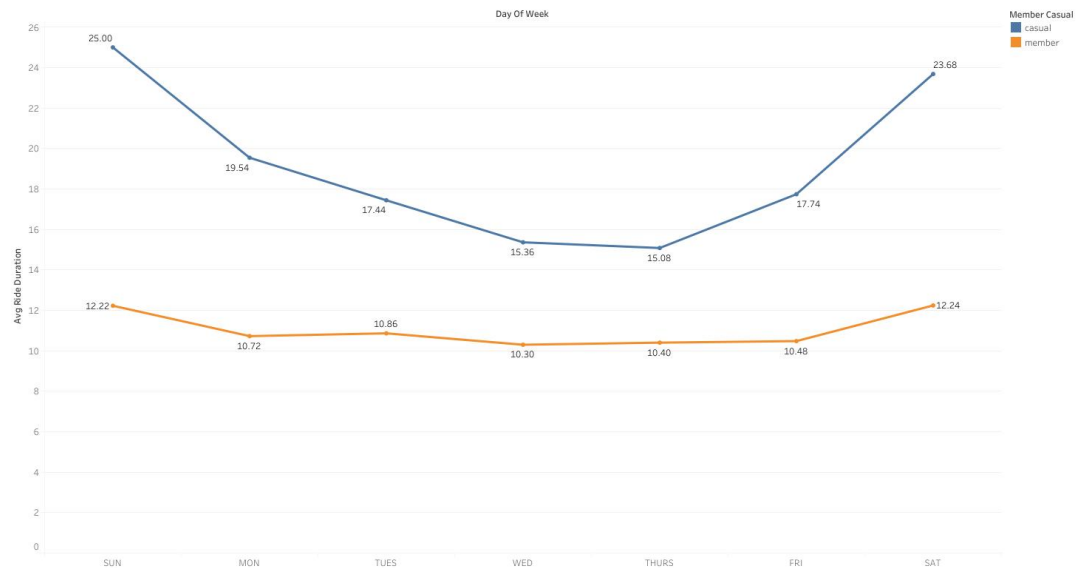

```sql
-- compare average ride_length per hour between member_casual
SELECT EXTRACT(HOUR FROM started_at) AS hour_of_day, member_casual, AVG(ride_length)
AS avg_ride_length
FROM `my-first-project-452000.2024_divvy_ridedata.2024_tota_cleaned_2`
GROUP BY hour_of_day, member_casual
ORDER BY hour_of_day
```

Average Ride Length Pour Hour

The trend of sum of Avg Ride Length for Hour Of Day. Color shows details about Member Casual.

Share

1.How do annual members and casual riders use Cyclistic bikes differently?

As illustrated in the figures above, casual riders have higher "average_ride_length", but with less total "number_of_rides". Casual riders tend to have a higher average ride length, indicating strong demand and high potential for conversion to membership.

2. Why would casual riders buy Cyclistic annual memberships?

To leverage this, we can target advertising in areas with high casual rider density to increase awareness of membership benefits (There are location information in the original data file, but for this case study, there are more than 50% location info missing, so I did not include in my analysis this time). Additionally, offering discounts based on average ride duration would appeal to high-usage casual riders, making membership more cost-effective and attractive to them.

3. How can Cyclistic use digital media to influence casual riders to become members?

Focus digital media efforts on areas with high casual rider activity , and offer membership discounts based on "average_ride_length" to incentive conversion.

Act

Based on current data analysis, I recommend the following promotional strategy to encourage membership sign-ups and retain existing members:

Offer tiered rewards based on members' annual average ride length:
Tier 1: Ride over XXX km/year – receive [e.g., partial membership discount / gift / credit]
Tier 2: Ride over XXXX km/year – receive [e.g., larger discount / premium reward]
Tier 3: Ride over XXXXX km/year – receive [e.g., full discount / exclusive gift / high-value credit]
This performance-based incentive encourages regular use and strengthens member loyalty.