**Capstone Proyect Google Data Analyst.**

Case Study 1 - How Does a Bike-Share Navigate Speedy Success?

Business Task:

The business has already identified 2 main classes of customers. One class corresponds to casual riders who adquire either single ride or daily passes, and members, which are those clients that use the service regularly and have annual ridership passes. As financial analysts concluded in their own analysis, annual riderships are far more profitable, and marketing leader Lily Moreno thinks that it is better to trying to convert casual riders to annual members because these casual riders already have the experience with the program. So, what the business task is to design a marketing campaign aimed at those customers that are more likely to change from casual ridership plans to member plans.

As working in this case as an analyst, my insights should stablish which are the main characteristics of the clients that are more likely to switch from a regular to a annual plan. These insights will allow the company to aim the campaign to the right group of customers, thus optimizing the marketing budget and improving the conversion of the campaign.

The key stakeholders we must consider in the project are:

1. Managers which are interested in improving the company’s profit.
2. Clients, which are the target of the campaign and they could benefit of adquiring a more convenient product.
3. Society. As current customers are convinced to use more the service, this could result in a decrease of the usage of contaminating transport means such as cars and therefore reducing traffic, which is an important concern in most cities.

Data preparation.

Data for this case is located on a AWS bucket. There is available data from the past up to 2020. We will use, as stated the guide the data available for the last twelve months. This includes from nov-21 to oct-22.

In order to work more easly, a copy of the data will be downloaded and stored locally. This data is historical data so It would not be modified anytime. The data is public available under a licence. I read carefully the licence and I understood the conditions and conditions. Data is secure because it resides in AWS which has strong security measres and the copy used for my own analysis will be kept until the analysis is fully done. As the data is huge, in order to perform sorting and filtering I will use bigCloud, storing the data in a google cloud Bucket.

I checked data integrity and consistency by manually exploring data. The data has no issues with bias as it represents all the trips recorded by the company during a certain period of time and they come from Motivate Inc, a company with a known reputation which operates more than 57% of bike shared rides as stated in their web. After initial exploration I found no special problems with this piece of data.

Analyzing the ROCCC of the data we can establish that:

**Reliability:** The data is reliable as it represents all the rides and it is supported by the city of Chicago.

**Originality:** Data is original as it contains raw records of bike trips.

**Comprehensiveness:** A limitation is that there is no personal information about clients.

**Current:** This data is from the past 12 months, not including the current month so the newest records are from less than 1 month ago. For this specific analysis the data is current.

**Cited:** This data is cited by google to be used in it’s capstone project.

In order to process the data I will chose to use SQL, because it’s a very big amount of raw data and SQL allows to process it fast and efficiently.

To look data at a Glance, we perform this SQL search for each dataset.

SELECT \*

FROM `authentic-host-368023.bike\_sharing.2021-11`

LIMIT 10

Looking at the data I can see that some of the records have missing information on the station. Some of them lack the station name, while in other records the latitude or longitude. As latitude and longitude are unique to the station we can use the information on one field to fill the other.

First, we need to identify the missing information. Data are splitted in 12 different tables ( 1 per month). In order to simplify queries we concatenate the 12 tables with the following command.

SELECT \*

FROM

`authentic-host-368023.bike\_sharing.2021-11`

UNION DISTINCT

SELECT \*

FROM

`authentic-host-368023.bike\_sharing.2021-12`

UNION DISTINCT

SELECT \*

FROM

`authentic-host-368023.bike\_sharing.2022-1`

UNION DISTINCT

.

.

.

.

SELECT \*

FROM

`authentic-host-368023.bike\_sharing.2022-10`

I set the destination for the query to a new table called bike\_sharing\_last\_year.

Now that I have all data together I start by counting the number of records and the number of records with missing values.

SELECT COUNT(\*)

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`

Results are that we have **5755694** records.

We count how many null values are.

SELECT COUNT(\*)

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`

WHERE start\_station\_name IS NULL

OR start\_station\_id IS NULL

OR end\_station\_name IS NULL

OR end\_station\_id IS NULL

OR start\_lat IS NULL

OR start\_lng IS NULL

OR end\_lat IS NULL

OR end\_lng IS NULL

The result is **1345256**

And the percentage

SELECT

(

SELECT

COUNT(\*)

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`

WHERE start\_station\_name IS NULL

OR start\_station\_id IS NULL

OR end\_station\_name IS NULL

OR end\_station\_id IS NULL

OR start\_lat IS NULL

OR start\_lng IS NULL

OR end\_lat IS NULL

OR end\_lng IS NULL

)

/

(SELECT

COUNT(\*)

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`

) \* 100

More than 23% of the records have missing values. Of those, only 5000 have missing either latitude or longitude. We can use the other records to fill this gaps later. By now I will create 2 new columns to easy the analysis. This columns are ride\_length and day of week, updating the current query.

SELECT \*, TIMESTAMP\_DIFF(TIMESTAMP(ended\_at),TIMESTAMP(started\_at),MINUTE) AS ride\_length, EXTRACT(DAYOFWEEK FROM started\_at)

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`

Latitude and longitude coordinates are not exactly the same for the same stations as I can see if running the following query.

#try to match coordintes with stations

SELECT DISTINCT

CONCAT(start\_lat, '\_', start\_lng) AS start\_coord,

start\_station\_name AS name,

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`

WHERE start\_station\_name IS NOT NULL

LIMIT 10

One possibilty is to truncate the coordinates to the first 3 numbers after the colon, but this could potentially bring some errors. So we are going to just not consider records that lack the satation name when this information is needed.

|  |  |
| --- | --- |
| membership\_type | total\_members |
| member | 3402661 |
| casual | 2353033 |

We can see that members have the majority of the rides (more than 60%). We calculate average bike length ride and mode of day of week (the day of the week with more rides) for both clases of memberships. We use the following query to do so.

WITH bike\_share\_last\_year\_calculation

AS

(SELECT \*,

TIMESTAMP\_DIFF(TIMESTAMP(ended\_at),TIMESTAMP(started\_at),MINUTE)

AS ride\_length,

EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_trip,

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`)

SELECT

member\_casual AS membership\_type, COUNT(member\_casual) AS total\_members, (day\_of\_trip,1) AS mode

FROM bike\_share\_last\_year\_calculation

group by member\_casual

|  |  |  |  |
| --- | --- | --- | --- |
| **Membership\_type** | **Count** | **Mode.value** | **Mode.count** |
| casual | 2353033 | 7 | 485281 |
| member | 3402661 | 5 | 532670 |

We can see that for casual members, the day of the week with most rides is Saturday and for members is Thursday. This has certain logic as members tend to go to work by bike while casual members use bikes for leisure, mainly on weekends. Let’s explore this further.

WITH bike\_share\_last\_year\_calculation

AS

(SELECT \*,

TIMESTAMP\_DIFF(TIMESTAMP(ended\_at),TIMESTAMP(started\_at),MINUTE)

AS ride\_length,

EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_trip,

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`)

SELECT

member\_casual AS membership\_type, COUNT(member\_casual) AS total\_trips, day\_of\_trip, ROUND(AVG(ride\_length),2) AS Avg\_ride

FROM bike\_share\_last\_year\_calculation

group by member\_casual, day\_of\_trip

ORDER BY day\_of\_trip, membership\_type

|  |  |  |  |
| --- | --- | --- | --- |
| **member\_casual** | **total\_trips** | **day\_of\_rip** | **Avg\_ride** |
| casual | 397079 | 1 | 33.36 |
| member | 395852 | 1 | 13.59 |
| casual | 284971 | 2 | 28.8 |
| member | 489686 | 2 | 11.81 |
| casual | 264399 | 3 | 25.53 |
| member | 523711 | 3 | 11.64 |
| casual | 275394 | 4 | 24.42 |
| member | 529657 | 4 | 11.57 |
| casual | 306950 | 5 | 24.98 |
| member | 532670 | 5 | 11.76 |
| casual | 338959 | 6 | 27.42 |
| member | 476695 | 6 | 11.99 |
| casual | 485281 | 7 | 32.14 |
| member | 454390 | 7 | 13.69 |

As we can see, for both casual and member clients, the longest trips are done on weekends (days 1 and 7). Also, in the weekdays, trips are done mainly by members while on weekends, this are mainly done by casual members.

Now I will look at the geographical data, by analyzing how are trips distributed in the stations. I want to identify which are the stations that have more casual members, so marketing can target the campaign towards clients residing near those stations.

SELECT

start\_station\_name, COUNT(start\_station\_name)

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`

GROUP BY start\_station\_name

There are 1640 start stations. On the 10 stations have the most casual rides, so we can think they have more potential clients to guide the campaign.

SELECT

start\_station\_name, COUNT(start\_station\_name) as num\_trips\_casual

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`

WHERE member\_casual = 'casual'

GROUP BY start\_station\_name

ORDER BY num\_trips\_casual DESC

LIMIT 10

|  |  |  |
| --- | --- | --- |
|  | **start\_station\_name** | **num\_trips\_casual** |
| 1 | Streeter Dr & Grand Ave | 58383 |
| 2 | DuSable Lake Shore Dr & Monroe St | 32598 |
| 3 | Millennium Park | 25990 |
| 4 | Michigan Ave & Oak St | 25327 |
| 5 | DuSable Lake Shore Dr & North Blvd | 23792 |
| 6 | Shedd Aquarium | 20451 |
| 7 | Theater on the Lake | 18574 |
| 8 | Wells St & Concord Ln | 16392 |
| 9 | Dusable Harbor | 14270 |
| 10 | Clark St & Armitage Ave | 13964 |

So now that we have a broad idea of this data, I will extract a table to do more advanced calculations an graphics. I will use R to do so as it is the best tool for both data manipulation and graphics. I will consider this variables:

Start\_station

Day of week

Avg ride length – As shorter rides are more likely to be done by members.

I run the following query to get this information.

WITH bike\_share\_last\_year\_calculation

AS

(SELECT \*,

TIMESTAMP\_DIFF(TIMESTAMP(ended\_at),TIMESTAMP(started\_at),MINUTE)

AS ride\_length,

EXTRACT(DAYOFWEEK FROM started\_at) AS day\_of\_trip,

FROM `authentic-host-368023.bike\_sharing\_last\_year.bike\_sharing\_last\_year`)

SELECT

start\_station\_name, COUNT(start\_station\_name) AS num\_trips\_per\_station, day\_of\_trip, member\_casual, ROUND(AVG(ride\_length),2) AS avg\_ride\_length

FROM

bike\_share\_last\_year\_calculation

GROUP BY start\_station\_name,day\_of\_trip,member\_casual

This analysis is continued in R and published as an Rmarkdown document.