ANÁLISIS DE CLIENTES DE “CYCLISTIC”

## FECHA INICIO: 16 de marzo de 2023

## Objetivo:

El Objetivo del proyecto es analizar a los clientes de la empresa, diferenciados entre los que pagan la membresía anual y los clientes ocasionales, para aumentar la tasa de clientes con membresía anual. Las conclusiones obtenidas del análisis se utilizarán para dirigir la campaña de marketing de la empresa hacía la conversión de los clientes ocasionales en clientes con membresía.

# Interesados:

La directora de marketing de la empresa.

El equipo de análisis de marketing.

El equipo ejecutivo de la empresa.

## Fuentes de datos:

<https://divvy-tripdata.s3.amazonaws.com/index.html>

Se analizan los 12 últimos meses de los registros de los clientes de la empresa. Desde enero de 2022 hasta diciembre de 2022.

## Preparación de los datos:

En esta sección se exploran los datos para conocer como están estructurados, obtener los datos y reunirlos en una misma tabla, además se realizan los cálculos para crear las nuevas columnas.

Lo primero es cargar los datos en Excel a partir de los archivos .csv como primera toma de contacto. Vemos que hay muchas filas y que no vamos a poder trabajar los datos con Excel.

Al ser archivos con tantos datos se trabajan con SQL, utilizando BigQuery de Google. El primer paso es crear un *database* llamado “ciclystic”, a continuación, se crean una tabla para cada mes importando el archivo csv a SQL, para los archivos de más de 100 Mb se cargan a través de Google Drive.

Las tablas contienen las siguientes variables:

|  |  |  |  |
| --- | --- | --- | --- |
| **fullname** | **mode** | **type** | **description** |
| ride\_id | NULLABLE | STRING | *ID único de cada viaje.* |
| rideable\_type | NULLABLE | STRING | *tipo de bicicleta usada en el viaje* |
| started\_at | NULLABLE | TIMESTAMP | *Fecha y hora de comienzo del viaje.* |
| ended\_at | NULLABLE | TIMESTAMP | *Fecha y hora de finalización del viaje* |
| start\_station\_name | NULLABLE | STRING | *Nombre de la estación desde la que comienza el viaje.* |
| start\_station\_id | NULLABLE | STRING | *ID único de la estación de comienzo del viaje.* |
| end\_station\_name | NULLABLE | STRING | *Nombre de la estación donde acaba el viaje.* |
| end\_station\_id | NULLABLE | STRING | *ID único de la estación donde acaba el viaje.* |
| start\_lat | NULLABLE | FLOAT | *Latitud de la posición de inicio.* |
| start\_lng | NULLABLE | FLOAT | *Altitud de la posición de inicio.* |
| end\_lat | NULLABLE | FLOAT | *Latitud de la posición final.* |
| end\_lng | NULLABLE | FLOAT | *Altitud de la posición final.* |
| member\_casual | NULLABLE | STRING | *Tipo de cliente que realiza el viaje* |

Para no realizar las mismas consultas en las 12 tablas para crear nuevas variables, lo primero es juntarlas en una sola tabla mediante la función UNION ALL.

El código es el siguiente:

SELECT

  ride\_id,

  rideable\_type,

  started\_at,

  ended\_at,

  start\_station\_name,

  start\_station\_id,

  end\_station\_name,

  end\_station\_id, start\_lat,

  start\_lng,

  end\_lat,

  end\_lng,

  member\_casual

 FROM `bloque4semana3.cyclistic.enero2022`

 UNION ALL

 SELECT

  ride\_id,

  rideable\_type,

  started\_at,

  ended\_at,

  start\_station\_name,

  start\_station\_id,

  end\_station\_name,

  end\_station\_id, start\_lat,

  start\_lng,

  end\_lat,

  end\_lng,

  member\_casual

 FROM `bloque4semana3.cyclistic.febrero2022`

 UNION ALL

 (... hasta unir las 12 tablas)

Una vez unidas tiene más de 5 millones de filas, así que se guarda la tabla como *datos2022* para no tener que repetir la unión en el código de futuras consultas.

A continuación, se crean las columnas con las nuevas variables *duración\_viaje*, *mes* y *dia\_semana.* Utilizando las funciones TIMESTAMP\_DIFF, DAYOFWEEK y MONTH, obtenemos el siguiente código:

SELECT ride\_id,

  rideable\_type,

  started\_at,

  ended\_at,

  ROUND (TIMESTAMP\_DIFF(ended\_at, started\_at, minute)) AS duracion\_viaje,

  EXTRACT(DAYOFWEEK FROM started\_at) AS dia\_semana,

  EXTRACT(MONTH FROM started\_at) AS mes,

  start\_station\_name,

  start\_station\_id,

  end\_station\_name,

  end\_station\_id, start\_lat,

  start\_lng,

  end\_lat,

  end\_lng,

  member\_casual

FROM `bloque4semana3.cyclistic.datos2022`;

La columna de *dia\_semana* muestra un valor numérico entre el 1 (domingo) y el 7 (sábado). La consulta se guarda en la base de datos como una tabla nueva llamada *tabladatos2022.*

## Limpieza de los datos:

Una vez que tenemos todos los datos en la misma tabla y con todas las variables creadas podemos empezar el proceso de limpieza.

Se comprueban el número de registros de *ride\_id* al ser la variable principal de los datos, por si existen duplicados:

SELECT DISTINCT COUNT (ride\_id) AS numero\_viajes

FROM `bloque4semana3.cyclistic.datos2022`;

El resultado nos da el mismo número de filas que tiene el archivo (5667717), por lo tanto, no hay ningún viaje repetido.

Además, queremos revisar que no haya valores nulos en las variables que usaremos para crear nuevas columnas que nos servirán en el posterior análisis. El código utilizado es el siguiente:

SELECT COUNT(\*) AS valores\_nulos,

FROM `bloque4semana3.cyclistic.datos2022`

WHERE ride\_id IS NULL;

El resultado de esta consulta es una columna con el número de valores nulos en los datos. Repitiendo la consulta para cada una de las distintas variables se obtiene el número de valores nulos en cada columna de la tabla. En las variables que nos interesan como son *ride\_id, rideable\_type, started\_at, ended\_at* y *member\_casual* no hay valores nulos. Por lo tanto, se considera que no es necesario eliminar ningún registro de los datos utilizados.

En las variables *start\_station\_name* y *end\_station\_name* sí que aparecen valores nulos, pero no los vamos a borrar porque son muchos registros. Se filtrarán las posteriores consultas para que no visualicen estos valores nulos.

Para la variable creada *duración\_viaje* procedemos a revisar los resultados obtenidos para encontrar posibles fallos en el cálculo debido a valores erróneos. El primer caso es contar todos los valores nulos o negativos en la duración del viaje:

SELECT COUNT(\*) AS viajes\_negativo,

FROM `bloque4semana3.cyclistic.tabladatos2022`

WHERE duracion\_viaje <= 0;

El resultado obtenido son 121089 filas con duración del viaje en negativo, con lo que estos datos no sirven para el análisis. Así que se eliminan de la tabla obtenida anteriormente:

DELETE FROM `bloque4semana3.cyclistic.tabladatos2022`

WHERE duracion\_viaje <= 0;

El siguiente paso es determinar los viajes máximos y mínimos, y la media de duración de todos los viajes:

SELECT ROUND(AVG(duracion\_viaje),1) AS media\_duracion,

    MAX (duracion\_viaje) AS max\_duracion,

    MIN (duracion\_viaje) AS min\_duracion,

FROM bloque4semana3.cyclistic.tabladatos2022;

El resultado de la consulta es que el tiempo medio de viaje es de 19.4 minutos, la duración máxima es de 41387 minutos (28 días), y la duración mínima es de 1 minuto. Como se puede observar hay valores extraños en los datos de la hora de finalización de los viajes, ya que aparecen datos de duración de varios días o incluso un mes entero. No se borran estos valores porque no podemos determinar a partir de que cantidad de tiempo se considera error.

## Análisis de los datos:

Realizando la misma consulta que en el paso anterior se obtiene la media de los viajes para cada tipo de cliente en la empresa:

SELECT ROUND(AVG(duracion\_viaje),2) AS media\_viaje,

  member\_casual

FROM `bloque4semana3.cyclistic.tabladatos2022`

GROUP BY member\_casual;

Obtenemos una media de duración de viaje para los miembros de 12,5 minutos, y para los usuarios casuales de 29,26 minutos.

Creamos 4 tablas a partir de las consultas de promedio de duración del viaje, por mes y por día de la semana, para miembros y clientes casuales.

SELECT

  ROUND(AVG(duracion\_viaje),2) AS media\_duracion,

  dia\_semana

FROM `bloque4semana3.cyclistic.tabladatos2022`

WHERE member\_casual = 'casual'

GROUP BY dia\_semana;

También se realiza la consulta con el número de viajes realizado para cada tipo de cliente:

SELECT member\_casual,

  COUNT(\*) AS num\_viajes

FROM `bloque4semana3.cyclistic.tabladatos2022`

GROUP BY member\_casual;

El resultado es: 3272486 viajes realizados por miembros y 2274142 viajes realizados por clientes casuales.

Para obtener el periodo del año en el que se realizan más viajes se crea una consulta para cada tipo de cliente con la cuenta del total de viajes por mes y por día de la semana.

SELECT COUNT(\*) AS num\_viajes,

dia\_semana

FROM `bloque4semana3.cyclistic.tabladatos2022`

WHERE member\_casual = 'member'

GROUP BY dia\_semana;

Se realiza la consulta las cuatro veces cambiando el tipo de cliente y la columna a filtrar.

Una vez obtenida toda la información del número de viajes, los promedios y las cantidades, pasamos a analizar el tipo de bicicleta que más utilizan los clientes:

SELECT rideable\_type,

COUNT(\*) AS viajes\_clientes

FROM `bloque4semana3.cyclistic.tabladatos2022`

WHERE member\_casual='member'

GROUP BY rideable\_type;

Como último paso procedemos a analizar las estaciones más populares entre los dos tipos de clientes:

SELECT

  start\_station\_name,

  COUNT(\*) AS viajes\_miembros

FROM `bloque4semana3.cyclistic.tabladatos2022`

WHERE member\_casual = 'member'

  AND start\_station\_name != 'null'

GROUP BY start\_station\_name

ORDER BY COUNT(\*) DESC

LIMIT 10;

Obtenemos las 10 estaciones desde donde empiezan los viajes para los clientes que son miembros. Realizamos esta consulta con los clientes casuales, y para las estaciones de finalización de los viajes.

Del mismo modo obtenemos las 10 estaciones donde finalizan más viajes:

SELECT

  start\_station\_name,

  COUNT(\*) AS viajes\_miembros

FROM `bloque4semana3.cyclistic.tabladatos2022`

WHERE member\_casual = 'member'

  AND start\_station\_name != 'NULL'

GROUP BY start\_station\_name

ORDER BY COUNT(\*) DESC

LIMIT 10;

Y por último unimos estas dos consultas para obtener una tabla con la suma de las estaciones en las que más bicicletas circulan:

SELECT

station\_name,

SUM (viajes\_miembros) AS total\_viajes

FROM `bloque4semana3.cyclistic.stations\_members`

GROUP BY station\_name

ORDER BY total\_viajes DESC

Obtenemos dos tablas con el número de viajes en las 10 estaciones más visitadas por las clientes ordenadas de mayor a menor.

## Visualización de los datos:

Para crear el informe con el resultado del análisis utilizaremos Power Bi, que se conecta con la base de datos en Google BigQuery y podemos cargar las tablas obtenidas de las consultas anteriores.

Visualizaciones de datos con Power BI:

Gráfico circular con el % de los viajes realizados por cada tipo de cliente.

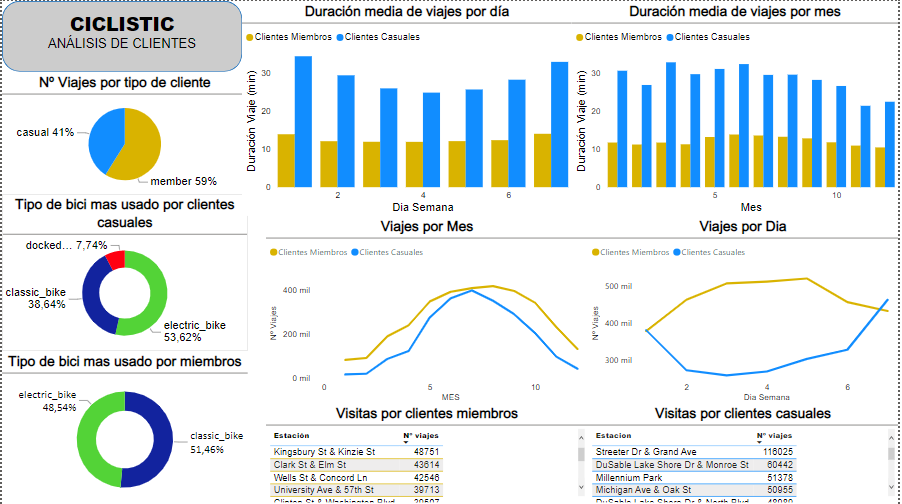
Gráfico de líneas con la variación del número de viajes por mes.

Gráfico de líneas con la variación del número de viajes en cada día de la semana.

Gráfico circular con la clase de bici más usado para los dos tipos de usuario.

Gráfico de barras con la duración media de los viajes para los tipos de clientes, en cada mes del año, y en cada día de la semana.

Tablas con las estaciones más visitadas para los dos tipos de clientes.



## Conclusiones:

Con el informe se pueden extraer las siguientes conclusiones sobre los clientes que utilizan las bicicletas de Cyclistic:

Más de la mitad de los viajes los realizan los clientes que son miembros. Usan casi por igual las bicicletas eléctricas y las clásicas, siendo estas un poco más usadas. Los clientes casuales prefieren las bicicletas eléctricas para sus viajes.

Los clientes casuales usan las bicicletas en las estaciones más cercanas a parques y zonas de ocio, mientras que los clientes miembros usan las bicicletas de estaciones más céntricas y cercanas a zona de trabajos y universidades. Los viajes que realizan los miembros se reparten más equitativamente entre las estaciones más visitadas, mientras que los usuarios casuales tienen preferencia por ciertas estaciones cercanas a zonas de ocio.

Los clientes miembros usan las bicicletas para viajes más cortos en todos los días de la semana que los clientes casuales. La duración media aumenta en ambos casos en los días de fin de semana. La media de duración de los viajes se mantiene superior para los clientes casuales durante todos los meses de año.

Ambos tipos de clientes usan más las bicicletas en la época del año que comprende entre Mayo y Septiembre, donde hay mejor tiempo para poder salir con la bicicleta. En cuanto a los días de la semana que más viajes se realizan, hay una gran diferencia entre los dos tipos de clientes. Los clientes casuales usan las bicicletas mayoritariamente los fines de semana, y los clientes miembros realizan más viajes en los días de entresemana o laborables.

Los clientes miembros están suscritos a cyclistic porque utilizan las bicicletas en el día a día para ir al trabajo o al instituto/universidad. Para aumentar el número de clientes miembros habría que ubicar nuevas estaciones cercanas a zonas de trabajo o de estudios, para que más gente pudiera optar por este servicio. Al usarlas más veces a la semana y durante periodos más cortos, prefieren las bicicletas clásicas ya que no requieren de recarga como las bicicletas eléctricas.