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

Paul

2026-01-15

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

Scenario

I am assigned as a junior data analyst working in the marketing analyst team at Cyclistic, a bike-share company in Chicago. The director of marketing believes the company's future success depends on maximizing the number of annual memberships. Therefore, my team wants to understand how casual riders and annual members use Cyclistic bikes differently. From these insights, your team will design a new marketing strategy to convert casual riders into annual members. But first, Cyclistic executives must approve your recommendations, so they must be backed up with compelling data insights and professional data visualizations.

Background

Cyclistic's finance analysts have concluded that annual members are much more profitable than casual riders. My manager, Lily Moreno, believes that maximizing the number of annual members will be key to future growth. Rather than creating a marketing campaign that targets all-new customers, Moreno believes there is a very good chance to convert casual riders into members.

Stakeholders

Primary stakeholder: Cyclistic executive team: The notoriously detail-oriented executive team will decide whether to approve the recommended marketing program.

Secondary stakeholders :

Lily Moreno: The director of marketing and your manager. Moreno is responsible for the development of campaigns and initiatives to promote the bike-share program. These may include email, social media, and other channels

Cyclistic marketing analytics team: I joined this team of data analysts six months ago. We are responsible for collecting, analyzing, and reporting data that helps guide Cyclistic marketing strategy and help Cyclistic achieve them.

Data Preparation

I will be using Cyclist Q1 2019 and Q1 2020 data sets for analysis, (because of Posit Rstudio free I cannot use large data sets) For the purposes of this case study, the datasets are appropriate and will enable you to answer the business questions. The data has been made available by Motivate International Inc.

under this licence create a link to the datasets? https://www.divvybikes.com/data-license-agreement

The data are collected internally using Cyclistic's own system. Then, the data are uploaded to a secured server for users to download. The data was downloaded and stored on my secured hard rive to insure privacy and security.

We can safely assume that the data is non bias, reliable, original, current, and cited as it is collected internally by Cyclistic. It is also comprehensive, since the data are formatted in .csv files.

Data Processing

I started exploring the data set on spreadsheet to get familiar with it. First i went checked for duplicates and empty rows, then I created new columns, the first was day of the week to have a view on which day users take bikes and the other column was the length of the trips i got them by comparing the start time of the ride and the end time of the ride.

I will continue the study on POSIT R Studio.

The spreadsheet function set number for each day and I changed them again to have the days and not numbers. 1 = Sunday 2 = Monday 3 = Tuesday 4 = Wednesday 5 = Thursday 6 = Friday 7 = Sunday

The libraries to read, clean and analyse the data.

```
library(tidyverse)  # Pour la manipulation et la visualisation
```

```
## ── Attaching core tidyverse packages ──────────────────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.4     ✔ readr     2.1.6
## ✔ forcats   1.0.1     ✔ stringr   1.6.0
## ✔ ggplot2   4.0.1     ✔ tibble    3.3.0
## ✔ lubridate 1.9.4     ✔ tidyr     1.3.1
## ✔ purrr     1.2.0
## ── Conflicts ────────────────────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(janitor)   # Pour gérer les dates facilement
```

```
##
## Attachement du package : 'janitor'
##
## Les objets suivants sont masqués depuis 'package:stats':
##
##     chisq.test, fisher.test
```

```r
library(lubridate)  # Pour nettoyer les noms de colonnes
library(ggplot2)
```

First I loaded the two data sets Divy_trips_2019 and Divy_trips_2020, I will set each set as df1 and df2

```r
Divvy_Trips_2019_Q1 <- read_csv("Divvy_Trips_2019_Q1.csv")
```

```
## Rows: 365069 Columns: 13
## ── Column specification ──────────────────────────────────────────────────────────────────────
─
## Delimiter: ","
## chr (7): start_time, end_time, from_station_name, to_station_name, usertype,...
## dbl (5): trip_id, bikeid, from_station_id, to_station_id, days_of_week_2019
## num (1): tripduration
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
Divvy_Trips_2020_Q1 <- read_csv("Divvy_Trips_2020_Q1.csv")
```

```
## Warning: One or more parsing issues, call `problems()` on your data frame for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)
```

```
## Rows: 426887 Columns: 16
## ── Column specification ──────────────────────────────────────────────────────────────────────
─
## Delimiter: ","
## chr  (8): ride_id, rideable_type, started_at, ended_at, start_station_name, ...
## dbl  (7): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## time (1): ride_length_2020
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
df1 <- Divvy_Trips_2019_Q1
df2 <- Divvy_Trips_2020_Q1
```

I changed the usertype column to have the same names for members ans casual users, plus I added a column annee to have a way to distinguish the data before joining in one data set.

```r
df1 <- df1  %>%
  mutate(usertype = recode(usertype,
                           "Subscriber" = "member",
                           "Customer" = "casual"))
df1 <- df1 %>% mutate(annee = "2019")
df2 <- df2 %>% mutate(annee = "2020")
```

Then, I joined the 2 data sets into one, and add some cleaning to make sure we don't have empty rows, I'have already checked the duplicates prior to this section.

```
bike_rides <- bind_rows(df1, df2)
bike_rides <- janitor::remove_empty(bike_rides, which = c("cols"))
bike_rides <- janitor::remove_empty(bike_rides, which = c("rows"))
```

I regrouped some of the columns together for analysis and visualizations.

```
bike_rides <- bike_rides %>%
  unite(Users, usertype, member_casual, sep = " ", na.rm = TRUE) #na rm to remove the NA that apears

bike_rides <- bike_rides %>%
  unite(ride_length, ride_length_2019, ride_length_2020, sep = " ", na.rm = TRUE)

bike_rides <- bike_rides %>%
  unite(day_of_the_week, day_of_the_week_2019, day_of_ther_week_2020, sep = " ", na.rm = TRUE)

bike_rides <- bike_rides %>%
  unite(days_of_the_week, days_of_the_week_2020, days_of_week_2019, sep = " ", na.rm = TRUE) #na rm to remove the NA that apears
```

I check that everything went right

```
str(bike_rides)
```

```
## tibble [791,956 × 26] (S3: tbl_df/tbl/data.frame)
##  $ trip_id          : num [1:791956] 21742443 21742444 21742445 21742446 21742447 ...
##  $ start_time       : chr [1:791956] "1/1/2019 0:04:37" "1/1/2019 0:08:13" "1/1/2019 0:13:23" "1/1/2019 0:13:45" ...
##  $ end_time         : chr [1:791956] "1/1/2019 0:11:07" "1/1/2019 0:15:34" "1/1/2019 0:27:12" "1/1/2019 0:43:28" ...
##  $ bikeid           : num [1:791956] 2167 4386 1524 252 1170 ...
##  $ tripduration     : num [1:791956] 390 441 829 1783 364 ...
##  $ from_station_id  : num [1:791956] 199 44 15 123 173 98 98 211 150 268 ...
##  $ from_station_name : chr [1:791956] "Wabash Ave & Grand Ave" "State St & Randolph St" "Racine Ave & 18th St" "California Ave & Milwaukee Ave" .
..
##  $ to_station_id    : num [1:791956] 84 624 644 176 35 49 49 142 148 141 ...
##  $ to_station_name  : chr [1:791956] "Milwaukee Ave & Grand Ave" "Dearborn St & Van Buren St (*)" "Western Ave & Fillmore St (*)" "Clark St & Elm
St" ...
##  $ Users            : chr [1:791956] "member" "member" "member" "member" ...
##  $ ride_length      : chr [1:791956] "0:06:30" "0:07:21" "0:13:49" "0:29:43" ...
##  $ days_of_the_week : chr [1:791956] "3" "3" "3" "3" ...
##  $ day_of_the_week  : chr [1:791956] "tuesday" "tuesday" "tuesday" "tuesday" ...
##  $ annee            : chr [1:791956] "2019" "2019" "2019" "2019" ...
##  $ ride_id          : chr [1:791956] NA NA NA NA ...
##  $ rideable_type    : chr [1:791956] NA NA NA NA ...
##  $ started_at       : chr [1:791956] NA NA NA NA ...
##  $ ended_at         : chr [1:791956] NA NA NA NA ...
##  $ start_station_name: chr [1:791956] NA NA NA NA ...
##  $ start_station_id  : num [1:791956] NA NA NA NA NA NA NA NA NA NA ...
##  $ end_station_name  : chr [1:791956] NA NA NA NA ...
##  $ end_station_id    : num [1:791956] NA NA NA NA NA NA NA NA NA NA ...
##  $ start_lat        : num [1:791956] NA NA NA NA NA NA NA NA NA NA ...
##  $ start_lng        : num [1:791956] NA NA NA NA NA NA NA NA NA NA ...
##  $ end_lat          : num [1:791956] NA NA NA NA NA NA NA NA NA NA ...
##  $ end_lng          : num [1:791956] NA NA NA NA NA NA NA NA NA NA ...
```

Here I change the type of the column ride_length from characters to a date format, so i can use it later for data visualization.

```
bike_rides$ride_length <- hms(bike_rides$ride_length)
```

```
## Warning in .parse_hms(..., order = "HMS", quiet = quiet): Some strings failed
## to parse
```

Analyse

First i studied the ride length on spreadsheet and compare the members and casual users of bikes. I used spreadsheets and a pivot table to do those calculations

member_casual	AVERAGE of ride_length_2020	MEDIAN of ride_length_2020	STDEV of ride_length_2020
casual	1:35:47	0:21:09	1.194345766
member	0:12:41	0:08:35	0.1592306171

member_casual	AVERAGE of ride_length_2019	MEDIAN of ride_length_2019	STDEV of ride_length_2019

member_casual	AVERAGE of ride_length_2019	MEDIAN of ride_length_2019	STDEV of ride_length_2019
casual	1:01:57	0:23:21	0.9694346869
member	0:13:54	0:08:21	0.2186102193

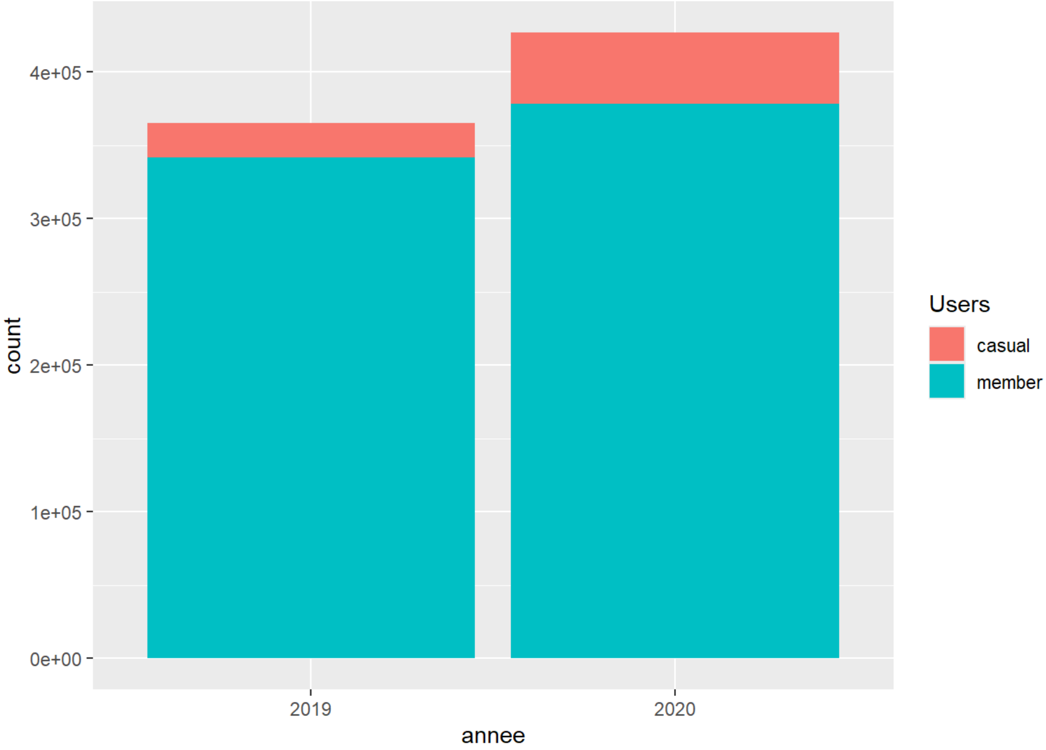
We can see with these values the difference of usage between casual and members usage of the bike rides, plus we can see that the usage between 2019 and 2020 is the same for each type of users. By comparing the ride length we can conclude that the casual member rides are way longer than the member average ride. We will see more with data visualizations.

Data Visualization

I used ggplot() to create a data visializations to have a better view of our data set.

Here we want to see if the proportion of users changed from on year to another based on the two data sets we are working on.

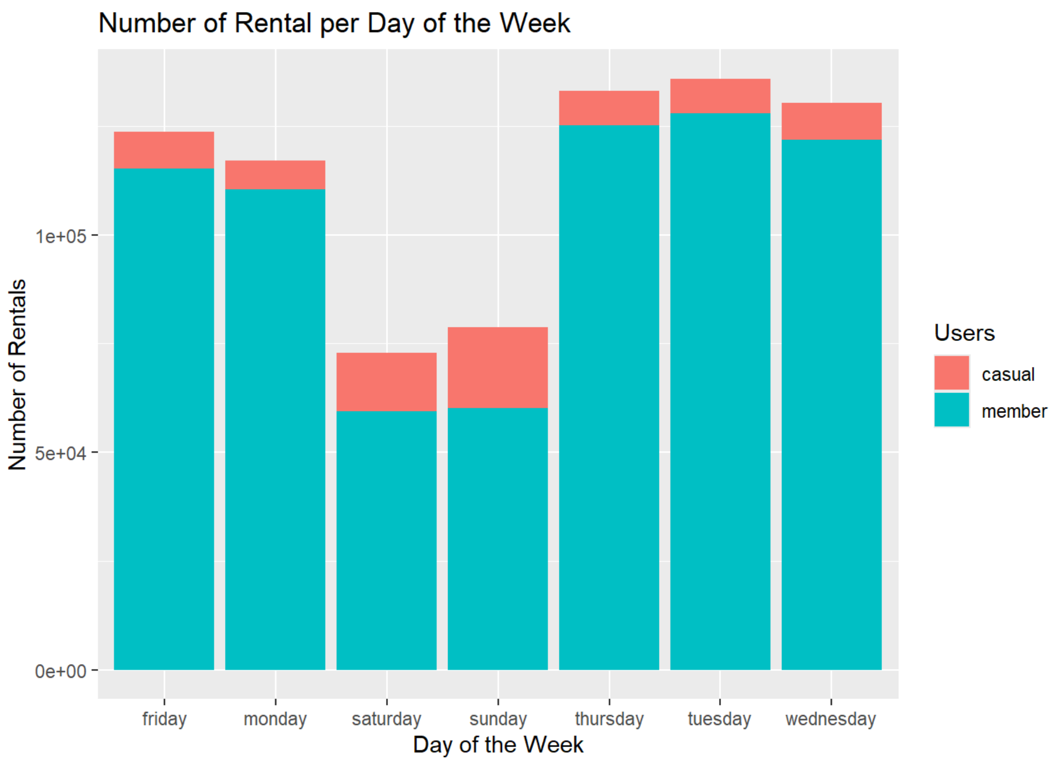
```
ggplot(data=bike_rides)+
  geom_bar(mapping = aes(x=annee, fill = Users))
```



We can observe a progression of users in one year, but still the proportion of users and members are alike. We can still see that the number of casual bike riders has more than doubled.

Here we will study the number of rental per day of the week and compare by type of users

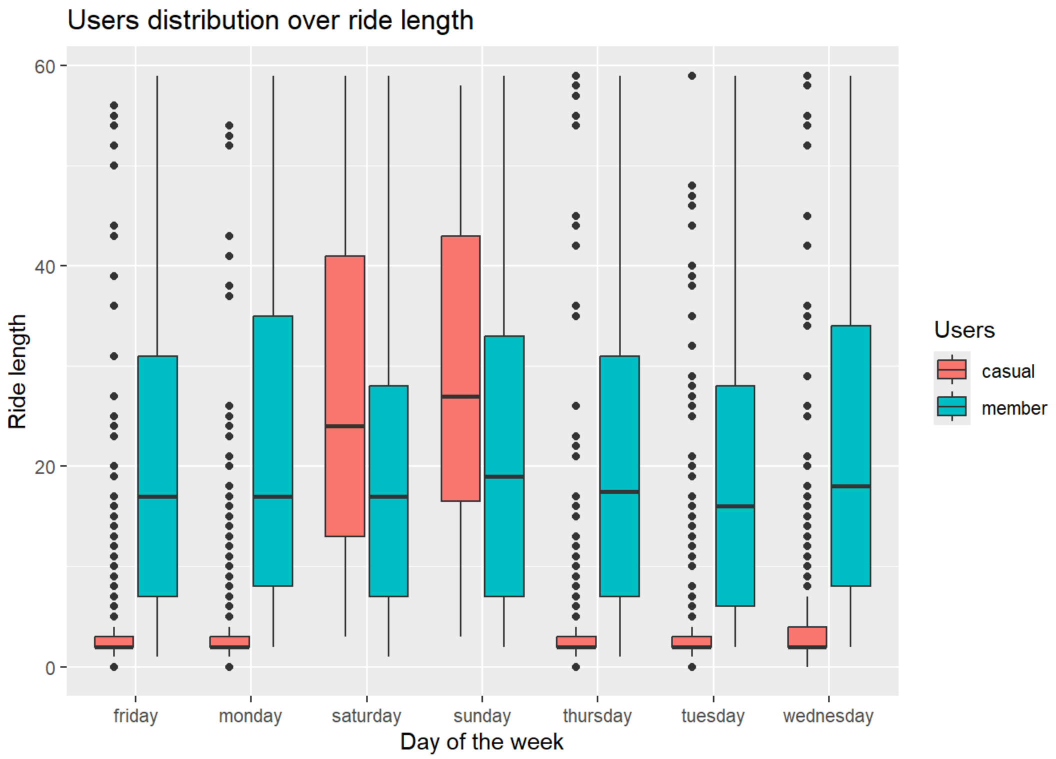
```
ggplot(data=bike_rides)+
  geom_bar(mapping = aes(x=day_of_the_week, fill = Users))+
  labs(title="Number of Rental per Day of the Week",
       x="Day of the Week",
       y="Number of Rentals",
  )
```



Here the data tells us members use bikes mostly during the working days and a bit less during the weekend. On the other hand casual users are renting bikes mostly during the week end.

```
ggplot(data = bike_rides, aes(x = day_of_the_week, y = (ride_length), fill = Users)) +
  geom_boxplot() +
  labs(title = "Users distribution over ride length",
       x = "Day of the week",
       y = "Ride length")
```

```
## Warning: Removed 784402 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```



The result of the analysis by comparing the usage of the members and casual users of the bikes. Casual members are using mostly bikes during the weekend, while members are using the bikes mostly during the working days.

Conclusion

The data is telling us the difference between the two types of users we compared them to get insights. The first type of users are members they can access a bike any time, during the analysis we learned they take bikes every day but mostly during the working days, probably to get to work and less during the weekend. Plus the biggest portion of users are members during the first quarter of the year, we don't have access to the rest of the years with the data sets we worked with.

On the other hand we learned that casual users are having longer rides sometimes way longer for example more than an hour and use them mostly during the weekend, I would guess to enjoy a bike ride during their free time. The rest of the days the proportion of casual riders is very low, in comparison members take less rides during the weekend but the difference with the working days isn't this big. We also so casual users are a small portion during the first quarter of the year, we can only guess for the rest but the proportion of them should grow with the sunny days the rest of the year. The casual users is growing group of users they more than doubled in one year at the same period, try to convert them to subscription is challenge for the company.

We could have more insights if we studied all the year and the past years and maybe see an evolution of the proportion of casual users. In the end casual users are most likely to use bikes in their free time or to enjoy a bike ride. The members are using the bikes more often and for smaller rides, i guess they use them to go around instead of a car.

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.