

## Introduction

The case study was part of Google Data Analytics Professional Certifications. The case study resolves around a fictional company Cyclistic bike-share. The director of marketing for Cyclistic bike-share would like to maximize the number of annual memberships by converting casual riders(non-members) to annual memberships. The director of marketing believes maximizing membership will help with company's future growth.

## Ask

Three questions will guide the future marketing program: 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?

## Preparing the data

In this section I downloaded the necessary libraries for the case study; dplyr,tidyverse, and lubridate. Next I downloaded all the data sets that were needed into R and combined them using rbind() under variable allset.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.1 --

## v ggplot2 3.3.3      v purrr   0.3.4
## v tibble  3.1.2      v stringr 1.4.0
## v tidyr   1.1.3      v forcats 0.5.1
## v readr   1.4.0

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##   date, intersect, setdiff, union
```

```
aprilset<-read.csv("202004-divvy-tripdata.csv")
mayset<-read.csv("202005-divvy-tripdata.csv")
juneset<-read.csv("202006-divvy-tripdata.csv")
julyset<-read.csv("202007-divvy-tripdata.csv")
augustset<-read.csv("202008-divvy-tripdata.csv")
sepset<-read.csv("202009-divvy-tripdata.csv")
octset<-read.csv("202010-divvy-tripdata.csv")
novset<-read.csv("202011-divvy-tripdata.csv")
decset<-read.csv("202012-divvy-tripdata.csv")
janset<-read.csv("202101-divvy-tripdata.csv")
febset<-read.csv("202102-divvy-tripdata.csv")
marset<-read.csv("202103-divvy-tripdata.csv")
```

```
allset<-rbind(aprilset,mayset,juneset,julyset,augustset,sepset,octset,novset,decset,janset,febset,marse
```

Here I used `str()` to get the number of rows and the number of variables/columns. `Summary()` to get the Class and Mode of each variables/columns. Here we see the data set has 10 variables/columns which are listed in the output.

```
str(allset)
```

```
## 'data.frame':   3489748 obs. of  13 variables:
## $ ride_id      : chr  "A847FADBBC638E45" "5405B80E996FF60D" "5DD24A79A4E006F4" "2A59BBDF5CDBA7
## $ rideable_type : chr  "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
## $ started_at   : chr  "2020-04-26 17:45:14" "2020-04-17 17:08:54" "2020-04-01 17:54:13" "2020-
## $ ended_at     : chr  "2020-04-26 18:12:03" "2020-04-17 17:17:03" "2020-04-01 18:08:36" "2020-
## $ start_station_name: chr  "Eckhart Park" "Drake Ave & Fullerton Ave" "McClurg Ct & Erie St" "Calif
## $ start_station_id : chr  "86" "503" "142" "216" ...
## $ end_station_name : chr  "Lincoln Ave & Diversey Pkwy" "Kosciuszko Park" "Indiana Ave & Roosevelt
## $ end_station_id   : chr  "152" "499" "255" "657" ...
## $ start_lat        : num  41.9 41.9 41.9 41.9 41.9 ...
## $ start_lng        : num  -87.7 -87.7 -87.6 -87.7 -87.6 ...
## $ end_lat          : num  41.9 41.9 41.9 41.9 42 ...
## $ end_lng          : num  -87.7 -87.7 -87.6 -87.7 -87.7 ...
## $ member_casual    : chr  "member" "member" "member" "member" ...
```

```
summary(allset)
```

```
##   ride_id      rideable_type      started_at      ended_at
## Length:3489748 Length:3489748 Length:3489748 Length:3489748
## Class :character Class :character Class :character Class :character
## Mode  :character Mode  :character Mode  :character Mode  :character
##
##
##
## start_station_name start_station_id end_station_name end_station_id
## Length:3489748 Length:3489748 Length:3489748 Length:3489748
```

```
## Class :character   Class :character   Class :character   Class :character
## Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##   start_lat      start_lng      end_lat      end_lng
## Min.   :41.64    Min.   : -87.87    Min.   :41.54    Min.   : -88.07
## 1st Qu.:41.88    1st Qu.: -87.66    1st Qu.:41.88    1st Qu.: -87.66
## Median :41.90    Median : -87.64    Median :41.90    Median : -87.64
## Mean   :41.90    Mean   : -87.64    Mean   :41.90    Mean   : -87.64
## 3rd Qu.:41.93    3rd Qu.: -87.63    3rd Qu.:41.93    3rd Qu.: -87.63
## Max.   :42.08    Max.   : -87.52    Max.   :42.16    Max.   : -87.44
##
##              NA's      :4738    NA's      :4738
## member_casual
## Length:3489748
## Class :character
## Mode  :character
##
##
##
##
```

## Data Cleanup

The code in this section will drop all any NA values and remove any duplicate values with the `distinct()` function. Also the following rows were removed as they do not provide any input with the companies goals. The following rows were removed `rideable_type`, `start_lat`, `start_lng`, `end_lat`, and `end_lng`.

```
allset<-allset %>% drop_na() %>% distinct(.keep_all = TRUE) %>% select(-c(rideable_type,start_lat,start_lng,end_lat,end_lng))
```

## Preparing the Date

To prepare the date I made a column called `ride_length` to get the ride length of all the customers by getting difference of `ended_at` and `started_at`. The values in `ride_length` are in minutes. The customers were categorized with 4 values; subscriber, members, customer, and casual. I made all the subscriber values changed to member and customer values changed to casual. I made a column called `day_of_week`, this column has the days of the week based on dates in `started_at`. The values in this column are as follows 1=Sunday,2=Monday,3=Tuesday,4=Wednesday,5=Thursday,6=Friday,7=Saturday, and 8=Sunday.

```
allset <- allset %>% mutate(ride_length= difftime(ended_at,started_at,units="min"),member_casual = recode(
  , "Subscriber" = "member"
  , "Customer" = "casual")
, day_of_week= wday(started_at))%>% filter(ride_length > 0) #1=Sunday
head(allset)
```

```
##           ride_id      started_at      ended_at
## 1 A847FADB638E45 2020-04-26 17:45:14 2020-04-26 18:12:03
## 2 5405B80E996FF60D 2020-04-17 17:08:54 2020-04-17 17:17:03
## 3 5DD24A79A4E006F4 2020-04-01 17:54:13 2020-04-01 18:08:36
## 4 2A59BBD5CDBA725 2020-04-07 12:50:19 2020-04-07 13:02:31
## 5 27AD306C119C6158 2020-04-18 10:22:59 2020-04-18 11:15:54
```

```
## 6 356216E875132F61 2020-04-30 17:55:47 2020-04-30 18:01:11
##           start_station_name start_station_id
## 1           Eckhart Park                86
## 2       Drake Ave & Fullerton Ave        503
## 3       McClurg Ct & Erie St             142
## 4   California Ave & Division St        216
## 5           Rush St & Hubbard St        125
## 6 Mies van der Rohe Way & Chicago Ave    173
##           end_station_name end_station_id member_casual  ride_length
## 1 Lincoln Ave & Diversey Pkwy          152      member 26.81667 mins
## 2       Kosciuszko Park                499      member  8.15000 mins
## 3   Indiana Ave & Roosevelt Rd          255      member 14.38333 mins
## 4       Wood St & Augusta Blvd          657      member 12.20000 mins
## 5   Sheridan Rd & Lawrence Ave          323      casual 52.91667 mins
## 6   Streeter Dr & Grand Ave             35      member  5.40000 mins
##   day_of_week
## 1           1
## 2           6
## 3           4
## 4           3
## 5           7
## 6           5
```

## Analyze the Date

This code chunk will output a table with average, maximum, minimum of ride\_length categorized by member\_casual. We see in this table casual members do have a higher average ride length, while members do have a higher maximum ride length. Interestingly both causal and members have the same minimum ride length.

```
member_min_max_avg <- allset %>% group_by(member_casual) %>% summarize(avg_ride_length=mean(ride_length),
head(member_min_max_avg)
```

```
## # A tibble: 2 x 4
##   member_casual avg_ride_length max_ride_length min_ride_length
##   <chr>         <drtn>          <drtn>          <drtn>
## 1 casual      45.11344 mins    55683.88 mins    0.01666667 mins
## 2 member      15.92386 mins    58720.03 mins    0.01666667 mins
```

This table shows the average duration and numbers of ride for each day of the week. While casual members do have a higher average duration for each day; members have a higher number of rides but are not on the bikes for as long as casual member.

```
allset %>%
  mutate(day = wday(started_at, label = TRUE)) %>% #creates weekday field using wday()
  group_by(member_casual, day) %>% #groups by usertype and weekday
  summarise(number_of_rides = n() #calculates the number of rides and average
, average_duration = mean(ride_length)) %>% # calculates the average duration
  arrange(day)
```

## 'summarise()' has grouped output by 'member\_casual'. You can override using the '.groups' argument.

```
## # A tibble: 14 x 4
## # Groups:   member_casual [2]
##   member_casual day    number_of_rides average_duration
##   <chr>         <ord>         <int> <drtn>
## 1 casual      Sun           252589 50.83689 mins
## 2 member      Sun           255294 18.10020 mins
## 3 casual      Mon           144491 45.07004 mins
## 4 member      Mon           257574 15.12339 mins
## 5 casual      Tue           138738 40.52128 mins
## 6 member      Tue           274525 14.96723 mins
## 7 casual      Wed           151017 40.48319 mins
## 8 member      Wed           294769 15.05202 mins
## 9 casual      Thu           158383 43.23258 mins
## 10 member     Thu           289445 15.02532 mins
## 11 casual     Fri           198883 42.94239 mins
## 12 member     Fri           294737 15.55998 mins
## 13 casual     Sat           322416 47.05698 mins
## 14 member     Sat           310828 17.65312 mins
```

## Visualizing the Data

The graph below shows a bar chart that shows number of rides by members and casual. Here just like the table we can see members generally have a higher number of rides.

```
allset %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday) %>%
  ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge")+ggtitle("Number of rides by member vs casual")
```

## 'summarise()' has grouped output by 'member\_casual'. You can override using the '.groups' argument.

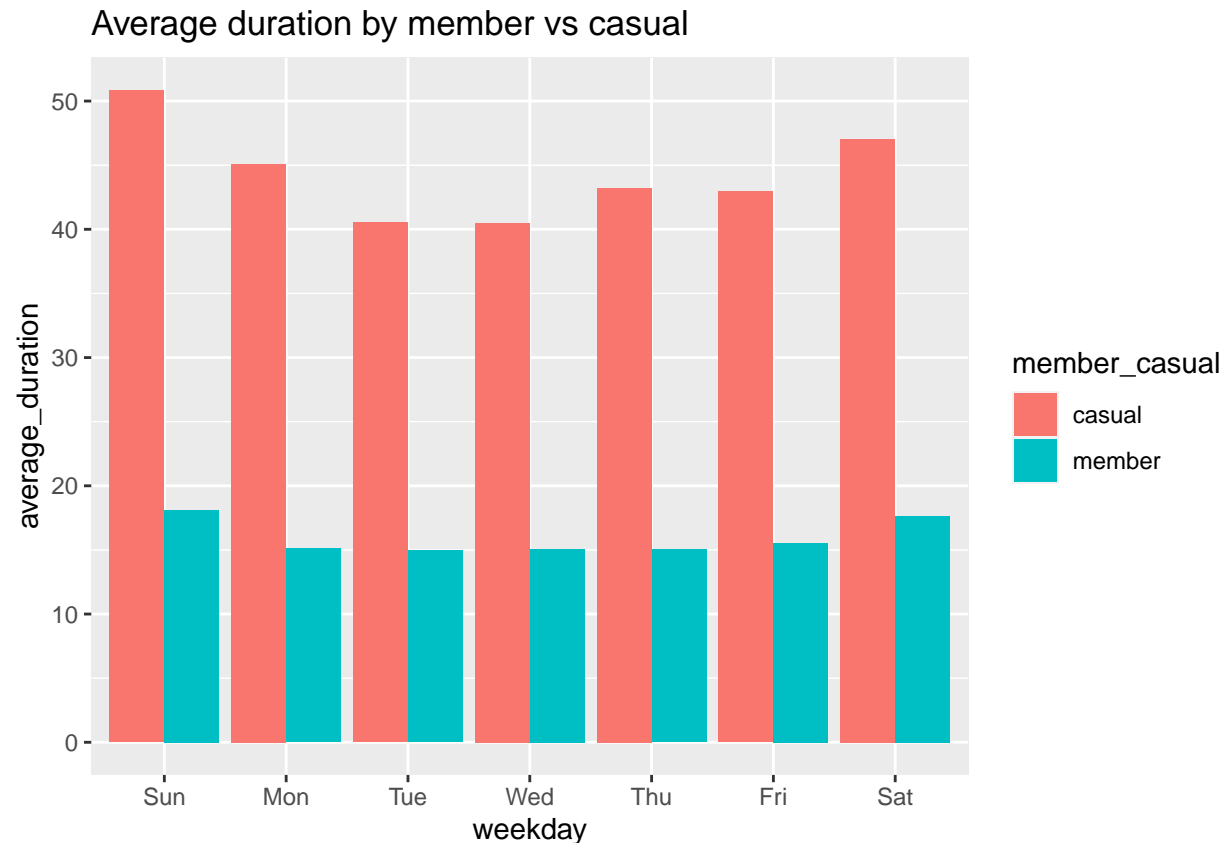


This graph shows another bar chart but for average duration by member and casual. Like the table we can see the casual members are on the bikes allot longer but not using them as frequently as members.

```
allset %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday) %>%
  ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge")+ggtitle("Average duration by member vs casual")
```

## 'summarise()' has grouped output by 'member\_casual'. You can override using the '.groups' argument.

## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.



## Conclusion

### Finding of the Data

Based on the data findings the casual members are using the bikes allot longer then members. This could be because casual members are buying single-ride passes and full-day passes in big groups who will not benefit from having an annual membership or people who just are visiting Chicago and need it for just one day. Members are people who live in Chicago and need the bikes for daily transportation.

### Getting casual riders to buy annual memberships

To get casual riders to buy annual memberships is to advertise to people who commute to the city for work. Cyclistic can have partnership with CTA or Metra to have their services included in their memberships as benefit for having a membership with CTA and Metra. Cyclistic can also have their bikes near the train stops to help promote their memberships.

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

Cyclistic can show their advertisement on social media platforms; Instagram and Youtube. Cyclistic can also have a promotion code for their causal riders when they sign up for an annual memberships that will give them a discount.