Cyclistic-Case-Study

Cody Hatch

11/16/2021

Introduction

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

Customers who purchase single-ride or full-day passes are referred to as casual riders. Customers who purchase annual memberships are Cyclistic members.

It's been concluded that annual members are much more profitable than casual riders. Although the pricing flexibility helps Cyclistic attract more customers, the team 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, the team believes there is a very good chance to convert casual riders into members. They note that casual riders are already aware of the Cyclistic program and have chosen Cyclistic for their mobility needs.

The executive team has set a clear goal: Design marketing strategies aimed at converting casual riders into annual members. In order to do that, however, the marketing analyst team needs to better understand how annual members and casual riders differ, why casual riders would buy a membership, and how digital media could affect their marketing tactics. The marketing team are interested in analyzing the Cyclistic historical bike trip data to identify trends.

I will be answering their questions using the six steps of data analytics taught throughout the Google Data Analytics course: ask, prepare, process, analyze, share, & act.

Ask

The main question this report is concerned with is:

How do annual members and casual riders use Cyclistic differently?

Prepare

To begin, the rider data for the last 12-13 months has been downloaded from the company and can be found here.

Note:

- The datasets have a different name because Cyclistic is a fictional company.
- The data has been made available by Motivate International Inc. under this license.
- Data-privacy issues prohibit the use of riders' personally identifiable information.

After being downloaded the data was stored locally as csv files in a data directory. The data was loaded and merged together into a single dataframe.

We'll want to start with a general overview of what the data looks like and dig deeper from there.

```
working_directory <- getwd()
data_directory <- "/Data/"
vec = cbind(working_directory, data_directory)
filedir <- paste(working_directory, data_directory, sep="")
file_names <- paste(filedir, dir(filedir), sep="")
merged_data <- do.call(rbind,lapply(file_names,read.csv))
str(merged_data)
```

```
## 'data.frame':    5767487 obs. of  13 variables:
##  $ ride_id           : chr  "ACB6B40CF5B9044C" "DF450C72FD109C01" "B6396B5
4A15AC0DF" "44A4AEE261B9E854" ...
##  $ rideable_type     : chr  "electric_bike" "electric_bike" "electric_bike
" "electric_bike" ...
##  $ started_at        : chr  "2020-10-31 19:39:43" "2020-10-31 23:50:08" "2
020-10-31 23:00:01" "2020-10-31 22:16:43" ...
##  $ ended_at          : chr  "2020-10-31 19:57:12" "2020-11-01 00:04:16" "2
020-10-31 23:08:22" "2020-10-31 22:19:35" ...
##  $ start_station_name: chr  "Lakeview Ave & Fullerton Pkwy" "Southport Ave
& Waveland Ave" "Stony Island Ave & 67th St" "Clark St & Grace St" ...
##  $ start_station_id  : chr  "313" "227" "102" "165" ...
##  $ end_station_name  : chr  "Rush St & Hubbard St" "Kedzie Ave & Milwaukee
Ave" "University Ave & 57th St" "Broadway & Sheridan Rd" ...
##  $ end_station_id    : chr  "125" "260" "423" "256" ...
##  $ start_lat         : num  41.9 41.9 41.8 42 41.9 ...
##  $ start_lng         : num  -87.6 -87.7 -87.6 -87.7 -87.7 ...
##  $ end_lat           : num  41.9 41.9 41.8 42 41.9 ...
```

```
##  $ end_lng           : num  -87.6 -87.7 -87.6 -87.7 -87.7 ...
##  $ member_casual     : chr  "casual" "casual" "casual" "casual" ...
```

```
summary(merged_data)
```

```
##    ride_id          rideable_type        started_at          ended_at
##  Length:5767487     Length:5767487     Length:5767487     Length:5767487
##  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:5767487     Length:5767487     Length:5767487     Length:5767487
##  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.   :-87.84   Min.   :41.51   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   :-87.65   Mean   :41.90   Mean   :-87.65
##  3rd Qu.:41.93   3rd Qu.:-87.63   3rd Qu.:41.93   3rd Qu.:-87.63
##  Max.   :42.08   Max.   :-87.52   Max.   :42.17   Max.   :-87.44
##                                   NA's   :5305    NA's   :5305
##  member_casual
##  Length:5767487
##  Class :character
##  Mode  :character
```

There’s a low count of missing values and they seem to all be concentrated in the latitude & longitude data - let’s get rid of those rows altogether.

```
data_no_na <- na.omit(merged_data)

original_len <- nrow(merged_data)
no_na_len <- nrow(data_no_na)
na_diff <- original_len - no_na_len

print(paste("Removed", na_diff, "null rows."))
```

```
## [1] "Removed 90824 null rows."
```

```
print(paste("That represented", na_diff/original_len * 100, "percent of the d
ata"))
```

```
## [1] "That represented 1.57475864271562 percent of the data"
```

Each ride has a distinct ride_id. If there are any duplicates for any of the data points, this is the most likely way to identify those duplicates.

```
data_no_duplicates <- data_no_na[!duplicated(data_no_na$ride_id), ]

no_dups_len <- nrow((data_no_duplicates))
no_dups_diff <- no_na_len - no_dups_len

print(paste("Removed", no_dups_diff, "duplicated rows"))

## [1] "Removed 208 duplicated rows"

print(paste("This represents another", no_dups_diff / original_len * 100, "pe
rcent of the original dataset"))

## [1] "This represents another 0.00360642338682341 percent of the original d
ataset"
```

I noticed the date columns were of the type 'character.' If we are going to glean any information from them we'll need to convert them to datetime format and parse through them that way.

```
data_no_duplicates$started_at <- as.POSIXct(data_no_duplicates$started_at, "%
Y-%m-%d %H:%M:%S")

data_no_duplicates$ended_at <- as.POSIXct(data_no_duplicates$ended_at, "%Y-%m
-%d %H:%M:%S")
```

Process

I think this is a good place to begin manipulating the data to add and subtract needed and unnecessary data points.

We have a start time and end time, but the information we really want is the length of the ride. So we'll do some simple maths and add a column for that data.

```
data_no_duplicates <- data_no_duplicates %>% mutate(ride_length = as.numeric(
data_no_duplicates$ended_at - data_no_duplicates$started_at)/60)

summary(data_no_duplicates$ride_length)

##      Min.    1st Qu.    Median      Mean    3rd Qu.      Max.
## -29049.97       6.97     12.35     20.68     22.33  55944.15
```

We can see some very interesting things about this ride_length data - we have a very large negative value for the minimum as well as a very large positive value for the maximum (almost 39 days long). Let's see about getting rid of some outliers.

```
tiles = quantile(data_no_duplicates$ride_length, seq(0, 1, by=0.02))
print(tiles)

##            0%            2%            4%            6%            8%
## -29049.966667      1.466667      2.533333      3.166667      3.666667
##           10%           12%           14%           16%           18%
##      4.116667      4.516667      4.916667      5.300000      5.666667
##           20%           22%           24%           26%           28%
##      6.033333      6.400000      6.766667      7.150000      7.516667
##           30%           32%           34%           36%           38%
##      7.900000      8.300000      8.700000      9.100000      9.516667
##           40%           42%           44%           46%           48%
##      9.950000     10.400000     10.850000     11.333333     11.833333
##           50%           52%           54%           56%           58%
##     12.350000     12.883333     13.466667     14.066667     14.700000
##           60%           62%           64%           66%           68%
##     15.366667     16.100000     16.866667     17.683333     18.566667
##           70%           72%           74%           76%           78%
##     19.533333     20.566667     21.716667     22.983333     24.366667
##           80%           82%           84%           86%           88%
##     25.916667     27.633333     29.600000     31.933333     34.833333
##           90%           92%           94%           96%           98%
##     38.500000     43.350000     51.083333     64.816667     93.016667
##          100%
##  55944.150000
```

0-100% Range -> 84,994 minutes 4-96% Range -> 62 minutes

I think we can limit our data to the middle 92% of the data and drop the outliers.

```
data_no_outliers <- data_no_duplicates %>%
    filter(ride_length > as.numeric(tiles['4%'])) %>%
    filter(ride_length < as.numeric(tiles['96%']))

print(paste("Removed", nrow(data_no_duplicates) - nrow(data_no_outliers), "ro
ws as outliners" ))

## [1] "Removed 455950 rows as outliners"
```

We have the date and time data, but there may be some information we can glean from the day of the week. Let's add that to our new set of data with the outliers removed.

```
data_no_outliers <- data_no_outliers %>% mutate(start_day = weekdays(as.Date(
data_no_outliers$started_at)))
```

Let's separate the day from what we'll call year_month. We'll keep year & month paired together because we have two of some months. We don't want them to be added together into a single month possibly skewing our analysis.

```
data_no_outliers <- data_no_outliers %>%
    mutate(year_month = paste(strftime(data_no_outliers$started_at,"%Y"),
                              "-", strftime(data_no_outliers$started_at,
                              "%m"),
                              paste("(",
                              strftime(data_no_outliers$started_at,"%b"),
                              ")",sep="")))

data_no_outliers <- data_no_outliers %>% mutate(day = day(as.Date(data_no_out
liers$started_at)))
```

Another time-related piece of information I feel might be insightful is the hour of the day they began their ride. Let's get that information into it's own column as well.

```
data_no_outliers <- data_no_outliers %>% mutate(hour_start = hour(data_no_out
liers$started_at))

## Warning in as.POSIXlt.POSIXct(x, tz = tz(x)): unknown timezone '%Y-%m-%d %
H:%M:
## %S'
```

I feel like we've cleaned the data up and added all the additional columns we'll need to start analyzing. Before we get started let's save the data as we have it so if this Rmd file is lost, we have a csv file to start the analysis with.

```
data <- data_no_outliers

write.csv(data, paste(filedir, "data_cleaned.csv", sep=""))
```

Analyze

General Breakdown

Let's see what the distribution of the data looks like between members and casual riders.

```
data %>% group_by(member_casual) %>% summarise(count=length(ride_id), '%' = (
length(ride_id) / nrow(data)) * 100)
```

```
## # A tibble: 2 x 3
##   member_casual   count   `%`
##   <chr>           <int> <dbl>
## 1 casual        2299348  44.0
## 2 member        2921157  56.0
```

So the makeup of customers is 44% casual riders and 56% members There are 27% more members than casual riders by sheer count.

Ride Length & Type

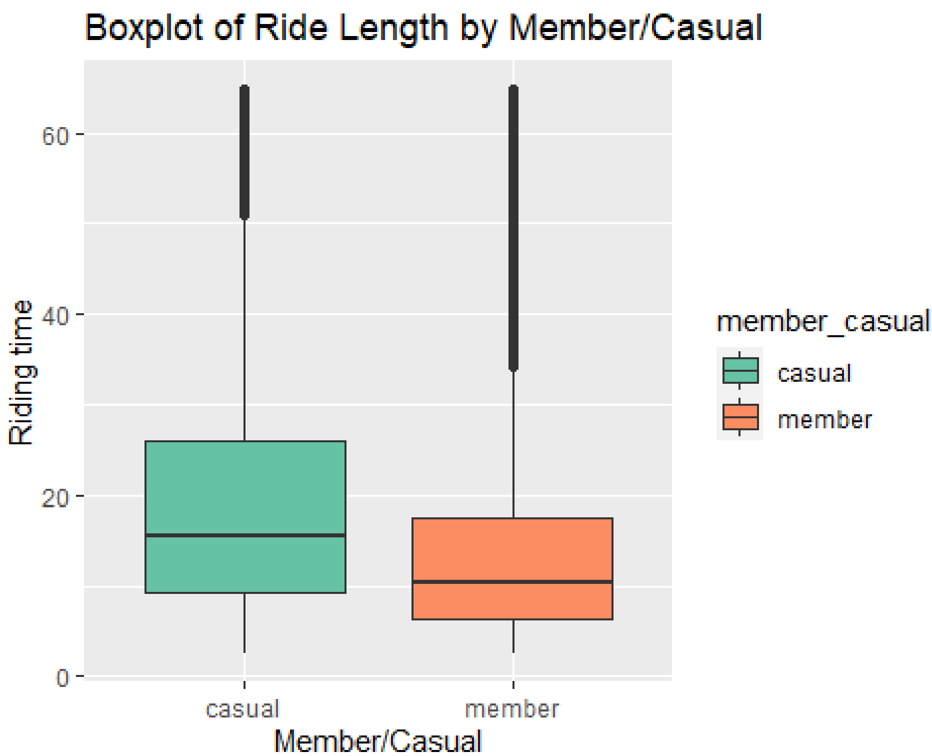
One of the possible differences could be in the way casual riders use the service vs members. Let's take a look at the ride length and bike types to see if there is information there.

```
data %>%
    group_by(member_casual) %>%
    summarise(mean = mean(ride_length),
              'Q1' = as.numeric(quantile(ride_length, .25)),
              'median' = median(ride_length),
              'Q3' = as.numeric(quantile(ride_length, .75)),
              'IR' = Q3 - Q1)
```

```
## # A tibble: 2 x 6
##   member_casual  mean    Q1 median    Q3    IR
##   <chr>         <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 casual         19.4  9.28   15.4  26.0  16.7
## 2 member         13.4  6.3    10.4  17.4  11.1
```

It looks like members tend to utilize

```
ggplot(data, aes(x=member_casual, y=ride_length, fill=member_casual)) +
    labs(x="Member/Casual", y="Riding time", title="Boxplot of Ride Length by
Member/Casual") +
    geom_boxplot() +
    scale_fill_brewer(palette="Set2")
```



We can kind of see that casual riders seem to have longer rides compared members.

Looking at the box plot is good and all, but I like hard numbers. Let's do some maths. A simple 2-Sample t-test should tell us with a bit more certainty if the means of these two groups are equal or not. The null hypothesis is that the means are equal and if the p-value is greater than the significance level (typically 0.05) means that we would fail to reject the null hypothesis. If the p-value is smaller, we reject the null hypothesis and can say that the means are NOT equal.

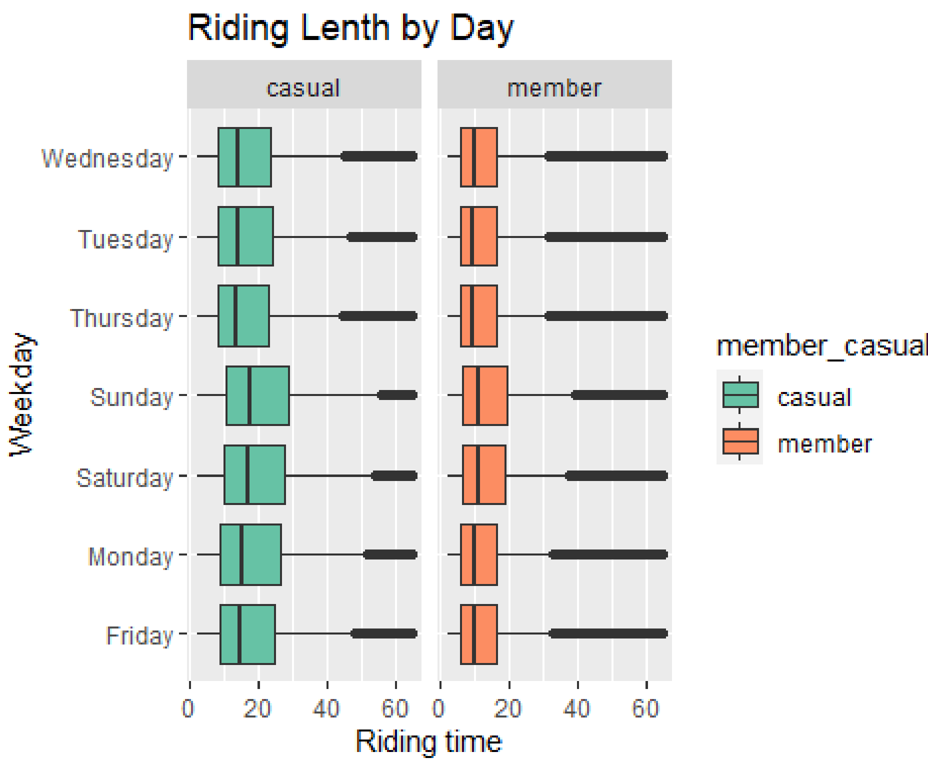
```
members <- data %>%
  filter(data$member_casual=='member')

causals <- data %>%
  filter(data$member_casual=='casual')

t.test(members$ride_length, causals$ride_length)

##
##  Welch Two Sample t-test
##
## data:  members$ride_length and causals$ride_length
## t = -564.75, df = 4074498, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -6.027401 -5.985709
## sample estimates:
```

```
## mean of x mean of y
##  13.43232  19.43887
```

Since the p-value is so small we can safely say, based on statistics, that the mean ride length of members is different from casual riders.

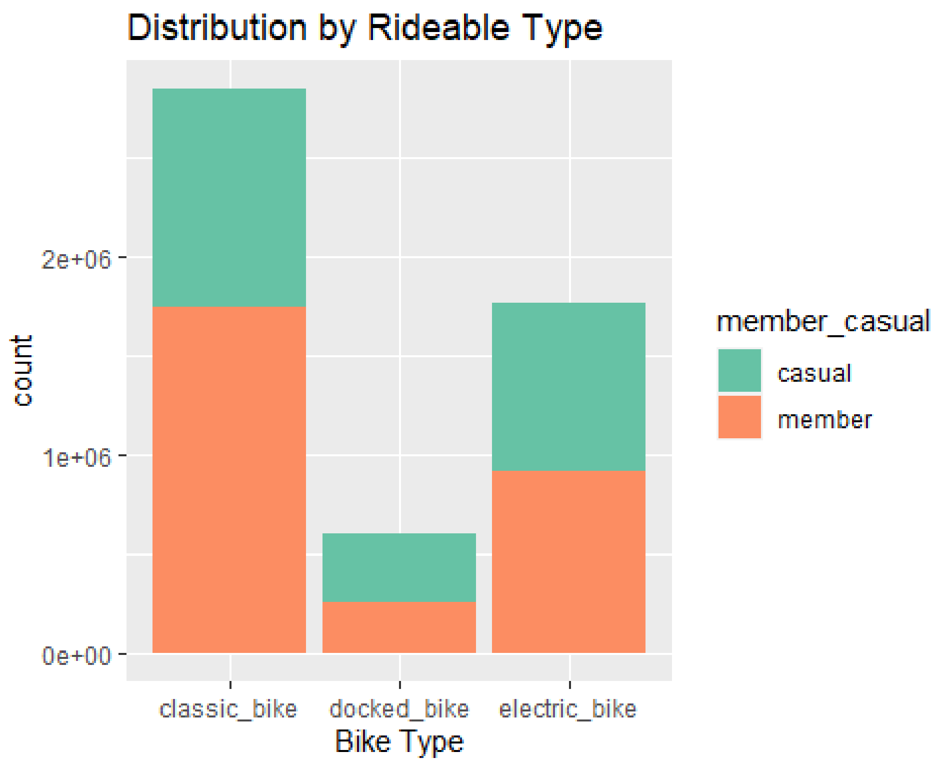
Let's see how their ride times differ throughout the week next.

```
ggplot(data, aes(x=start_day, y=ride_length, fill=member_casual, cex.axis=0.2
5)) +
    geom_boxplot() +
    facet_wrap(~ member_casual) +
    labs(x="Weekday", y="Riding time", title="Riding Lenth by Day") +
    scale_fill_brewer(palette="Set2") +
    coord_flip() # I tried normal axes and the labels overlapped
```



Okay, we can see that casual riders have a bit of an increase in ride time on the weekend, while members are more consistent.

```
data %>%
  ggplot(aes(rideable_type, fill=member_casual)) +
    geom_bar() +
    labs(x="Bike Type", title="Distribution by Rideable Type") +
    scale_fill_brewer(palette="Set2")
```



Interesting. Let's see the hard numbers.

```
data %>%
    group_by(rideable_type) %>%
    summarise(count = length(ride_id),
              '%' = (length(ride_id) / nrow(data)) * 100,
              'members_percent' = (sum(member_casual == "member") / length(ri
de_id)) * 100,
              'casual_percent' = (sum(member_casual == "casual") / length(rid
e_id)) * 100,
              'Difference' = members_percent - casual_percent)

## # A tibble: 3 x 6
##   rideable_type   count   `%` members_percent casual_percent Difference
##   <chr>           <int> <dbl>           <dbl>          <dbl>      <dbl>
## 1 classic_bike  2851387  54.6            61.2           38.8      22.3
## 2 docked_bike    598533  11.5            42.4           57.6     -15.2
## 3 electric_bike 1770585  33.9            52.1           47.9       4.29
```

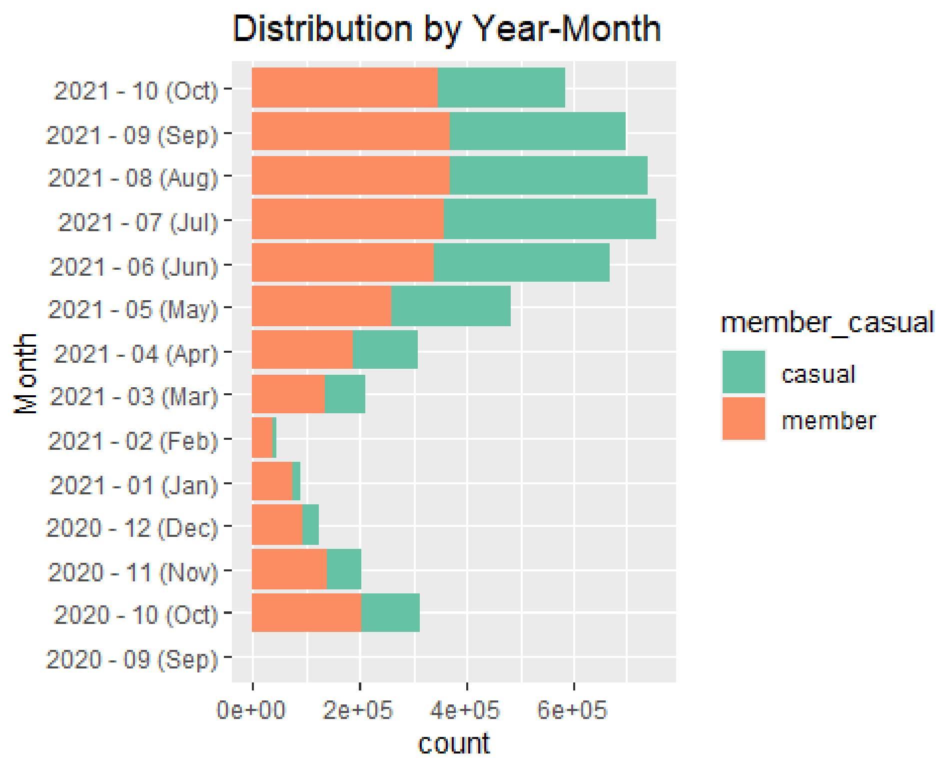
It appears that casual riders use the classic bike option significantly more than any of the other options.

Time Distribution

Let's take a look at the breakdown by year_month

```
data %>%
  ggplot(aes(year_month, fill=member_casual)) +
```

```
    geom_bar() +
    labs(x="Month", title="Distribution by Year-Month") +
    scale_fill_brewer(palette="Set2") +
    coord_flip() # I tried normal axes and the labels overlapped
```



The graph looks pretty, but lets see what the actual numbers look like so we can compare them

```
data %>%
    group_by(year_month) %>%
    summarise(count = length(ride_id),
              '%' = (length(ride_id) / nrow(data)) * 100,
              'members_percent' = (sum(member_casual == "member") / length(ri
de_id)) * 100,
              'casual_percent' = (sum(member_casual == "casual") / length(rid
e_id)) * 100,
              'Difference' = members_percent - casual_percent)

## # A tibble: 14 x 6
##    year_month       count      `%` members_percent casual_percent Differen
ce
##    <chr>            <int>    <dbl>           <dbl>          <dbl>      <db
l>
##  1 2020 - 09 (Sep)    104  0.00199            38.5           61.5     -23.1
##  2 2020 - 10 (Oct) 313228  6.00               65.2           34.8      30.3
##  3 2020 - 11 (Nov) 205356  3.93               68.8           31.2      37.5
##  4 2020 - 12 (Dec) 123098  2.36               77.5           22.5      55.1
```

```
## 5 2021 - 01 (Jan)  90680  1.74              81.6            18.4     63.2
## 6 2021 - 02 (Feb)  45944  0.880             80.6            19.4     61.2
## 7 2021 - 03 (Mar) 209925  4.02              65.1            34.9     30.1
## 8 2021 - 04 (Apr) 308600  5.91              61.4            38.6     22.7
## 9 2021 - 05 (May) 481140  9.22              53.8            46.2      7.5
0
## 10 2021 - 06 (Jun) 666125 12.8              50.9            49.1      1.8
4
## 11 2021 - 07 (Jul) 754635 14.5              47.6            52.4     -4.8
2
## 12 2021 - 08 (Aug) 738670 14.1              49.9            50.1     -0.1
93
## 13 2021 - 09 (Sep) 699593 13.4              52.8            47.2      5.6
7
## 14 2021 - 10 (Oct) 583407 11.2              59.7            40.3     19.4
```

It would seem the usage follows a seasonal pattern. We can also see how drastically the change in the casual rider usage changes with the seasons

Let's take a look at the breakdown by day

```
data %>%
    group_by(start_day) %>%
    summarise(count = length(ride_id),
              '%' = (length(ride_id) / nrow(data)) * 100,
              'members_percent' = (sum(member_casual == "member") / length(ri
de_id)) * 100,
              'casual_percent' = (sum(member_casual == "casual") / length(rid
e_id)) * 100,
              'Difference' = members_percent - casual_percent)
```

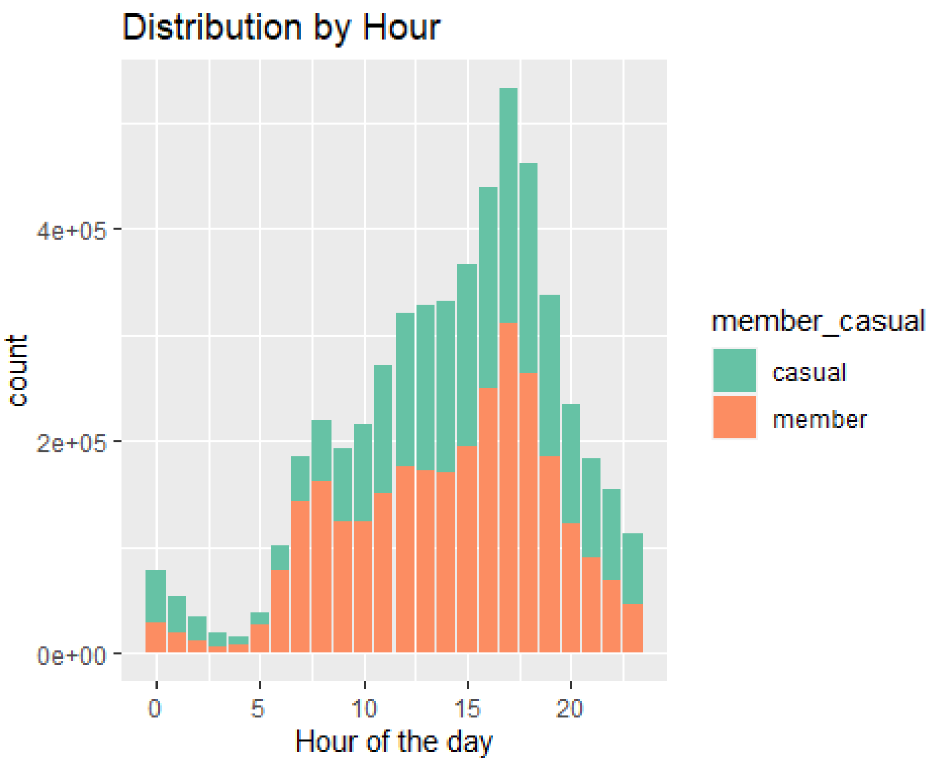
```
## # A tibble: 7 x 6
##   start_day  count   `%` members_percent casual_percent Difference
##   <chr>      <int> <dbl>           <dbl>          <dbl>      <dbl>
## 1 Friday    767568  14.7            56.1           43.9      12.2
## 2 Monday    639455  12.2            60.5           39.5      21.0
## 3 Saturday  940657  18.0            45.4           54.6      -9.13
## 4 Sunday    790512  15.1            46.0           54.0      -8.00
## 5 Thursday  700347  13.4            61.9           38.1      23.8
## 6 Tuesday   680285  13.0            63.5           36.5      27.0
## 7 Wednesday 701681  13.4            63.7           36.3      27.4
```

We can see a slight increase on the weekend (Friday-Sunday). In fact, we can see that casual riders overtake members on both Saturday and Sunday Otherwise, there's a consistent difference throughout the week.

Now I'm wondering how the data breaks down within the hour of the day. Let's take a look

```
data %>%
    ggplot(aes(hour_start, fill=member_casual)) +
```

```
    labs(x="Hour of the day", title="Distribution by Hour") +
    geom_bar() + scale_fill_brewer(palette="Set2")
```



Looking at the graph it looks like there's a significant peak in the afternoon from 3-6 and a less significant peak from 12-7

Share

Quick Summary of Findings

- Members make up the majority of users - 27% more than casual riders
- Bike usage spikes during warmer months and dips during colder months
- Casual riders overtake the number of members riding on the weekend
- Causal riders also increase their ride time on the weekends, while members stay consistent
- There's a spike in riding in the afternoons
- Classic bikes are the preferred type of bike

What Can We Conclude?

- Members are using bikes for more consistent things (work and/or exercise)
- Weekends are likely for recreational use & moreso by casual riders
- Temperature is a significant driver of usage

Act

Based on all of our findings and the overall conclusions I would recommend the following 3 steps to encourage casual riders to become members:

1. Ad campaign and special offers for members using bikes for commuting.
2. Ad campaign and special offers or benefits for members on the weekend.
3. Special offers during the colder months for members.