Cyclist-Bike-Share-Analysis

Sourabh Marne

2022-03-10

ASK PHASE

Q.1 What is the problem you are trying to solve?

Ans: Our main aim is to find marketing strategies to convert casual riders into annual members.

Q.2 How can your insights drive business decisions?

Ans: Our insights will help the marketing team increase the annual members.

PREPARE PHASE

1. The data is located in a kaggle dataset which is stored in csv files month by month.
2. The dataset contains entire population and there is no bias
3. The dataset ROCCCs as it is reliable, original, comprehensive, current and cited.

Loading the libraries

```
library(tidyverse)
```

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

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## -- 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
```

1. Tidyverse is used for data transformation.

2. Lubridate is used for working with dates and time efficiently.

PROCESS PHASE

Here we will prepare data for data analysis.

Reading the bikes data from all 12 months

```
df1 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202004-divvy-tripdata.csv")
df2 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202005-divvy-tripdata.csv")
df3 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202006-divvy-tripdata.csv")
df4 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202007-divvy-tripdata.csv")
df5 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202008-divvy-tripdata.csv")
df6 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202009-divvy-tripdata.csv")
df7 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202010-divvy-tripdata.csv")
df8 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202011-divvy-tripdata.csv")
df9 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202012-divvy-tripdata.csv")
df10 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202101-divvy-tripdata.csv")
df11 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202102-divvy-tripdata.csv")
df12 <- read.csv("C:/Users/hp/Desktop/Google Data Anlytics Capstone/CSV files/202103-divvy-tripdata.csv")
```

Binding all the dataframes together

```
rides <- rbind(df1,df2,df3,df4,df5,df6,df7,df8,df9,df10,df11,df12)
head(rides)
```

```
##            ride_id rideable_type          started_at            ended_at
## 1 A847FADBBC638E45   docked_bike 2020-04-26 17:45:14 2020-04-26 18:12:03
## 2 5405B80E996FF60D   docked_bike 2020-04-17 17:08:54 2020-04-17 17:17:03
## 3 5DD24A79A4E006F4   docked_bike 2020-04-01 17:54:13 2020-04-01 18:08:36
## 4 2A59BBDF5CDBA725   docked_bike 2020-04-07 12:50:19 2020-04-07 13:02:31
## 5 27AD306C119C6158   docked_bike 2020-04-18 10:22:59 2020-04-18 11:15:54
## 6 356216E875132F61   docked_bike 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 start_lat start_lng end_lat
## 1 Lincoln Ave & Diversey Pkwy            152   41.8964  -87.6610 41.9322
## 2             Kosciuszko Park            499   41.9244  -87.7154 41.9306
## 3  Indiana Ave & Roosevelt Rd            255   41.8945  -87.6179 41.8679
## 4      Wood St & Augusta Blvd            657   41.9030  -87.6975 41.8992
## 5  Sheridan Rd & Lawrence Ave            323   41.8902  -87.6262 41.9695
## 6     Streeter Dr & Grand Ave             35   41.8969  -87.6217 41.8923
##    end_lng member_casual
## 1 -87.6586        member
## 2 -87.7238        member
## 3 -87.6230        member
## 4 -87.6722        member
## 5 -87.6547        casual
## 6 -87.6120        member
```

Removing Duplicates

```
rides <- rides[!duplicated(rides$ride_id), ]
print(paste("Removed", nrow(rides) - nrow(rides), "duplicated rows"))
```

```
## [1] "Removed 0 duplicated rows"
```

Converting the “started_at” and “ended_at” columns from characters to Timestamps

```
rides$started_at <- lubridate::ymd_hms(rides$started_at)
rides$ended_at <- lubridate::ymd_hms(rides$ended_at)
```

Converting ride time in minutes

```
rides <- rides %>% mutate(ride_mins = as.numeric(rides$ended_at - rides$started_at) / 60)
summary(rides$ride_mins)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -29049.97      7.88     14.52     26.48     26.63  58720.03
```

Separating year and month

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

```
##  [1] "2020 - 04 (Apr)" "2020 - 05 (May)" "2020 - 06 (Jun)" "2020 - 07 (Jul)"
##  [5] "2020 - 08 (Aug)" "2020 - 09 (Sep)" "2020 - 10 (Oct)" "2020 - 11 (Nov)"
##  [9] "2020 - 12 (Dec)" "2021 - 01 (Jan)" "2021 - 02 (Feb)" "2021 - 03 (Mar)"
## [13] "2021 - 04 (Apr)"
```

Extracting the week days

```
rides <- rides %>% mutate(weekday = paste(strftime(rides$ended_at, "%u"), "-", strftime(rides$en
ded_at, "%a")))
unique(rides$weekday)
```

```
## [1] "7 - Sun" "5 - Fri" "3 - Wed" "2 - Tue" "6 - Sat" "4 - Thu" "1 - Mon"
```

Extracting the ride hour

```
rides$hour <- lubridate::hour(rides$ended_at)
unique(rides$hour)
```

```
##  [1] 18 17 13 11 14 10 15 16  3 12  9  8 20 19 21 23  7 22  6  1  5  0  2  4
```

ANALYZE PHASE

```
summary(rides)
```

```
##    ride_id          rideable_type        started_at
##  Length:3489539     Length:3489539     Min.   :2020-04-01 00:00:30
##  Class :character   Class :character   1st Qu.:2020-07-14 19:36:28
##  Mode  :character   Mode  :character   Median :2020-08-29 14:47:30
##                                        Mean   :2020-09-10 01:13:26
##                                        3rd Qu.:2020-10-20 18:07:35
##                                        Max.   :2021-03-31 23:59:08
##
##    ended_at                   start_station_name start_station_id
##  Min.   :2020-04-01 00:10:45   Length:3489539     Length:3489539
##  1st Qu.:2020-07-14 20:11:10   Class :character   Class :character
##  Median :2020-08-29 15:18:24   Mode  :character   Mode  :character
##  Mean   :2020-09-10 01:39:55
##  3rd Qu.:2020-10-20 18:21:47
##  Max.   :2021-04-06 11:00:11
##
##  end_station_name   end_station_id       start_lat       start_lng
##  Length:3489539     Length:3489539     Min.   :41.64   Min.   :-87.87
##  Class :character   Class :character   1st Qu.:41.88   1st Qu.:-87.66
##  Mode  :character   Mode  :character   Median :41.90   Median :-87.64
##                                        Mean   :41.90   Mean   :-87.64
##                                        3rd Qu.:41.93   3rd Qu.:-87.63
##                                        Max.   :42.08   Max.   :-87.52
##
##    end_lat         end_lng       member_casual        ride_mins
##  Min.   :41.54   Min.   :-88.07   Length:3489539     Min.   :-29049.97
##  1st Qu.:41.88   1st Qu.:-87.66   Class :character   1st Qu.:     7.88
##  Median :41.90   Median :-87.64   Mode  :character   Median :    14.52
##  Mean   :41.90   Mean   :-87.64                      Mean   :    26.48
##  3rd Qu.:41.93   3rd Qu.:-87.63                      3rd Qu.:    26.63
##  Max.   :42.16   Max.   :-87.44                      Max.   : 58720.03
##  NA's   :4737    NA's   :4737
##   year_month          weekday               hour
##  Length:3489539     Length:3489539     Min.   : 0.00
##  Class :character   Class :character   1st Qu.:12.00
##  Mode  :character   Mode  :character   Median :15.00
##                                        Mean   :14.52
##                                        3rd Qu.:18.00
##                                        Max.   :23.00
##
```

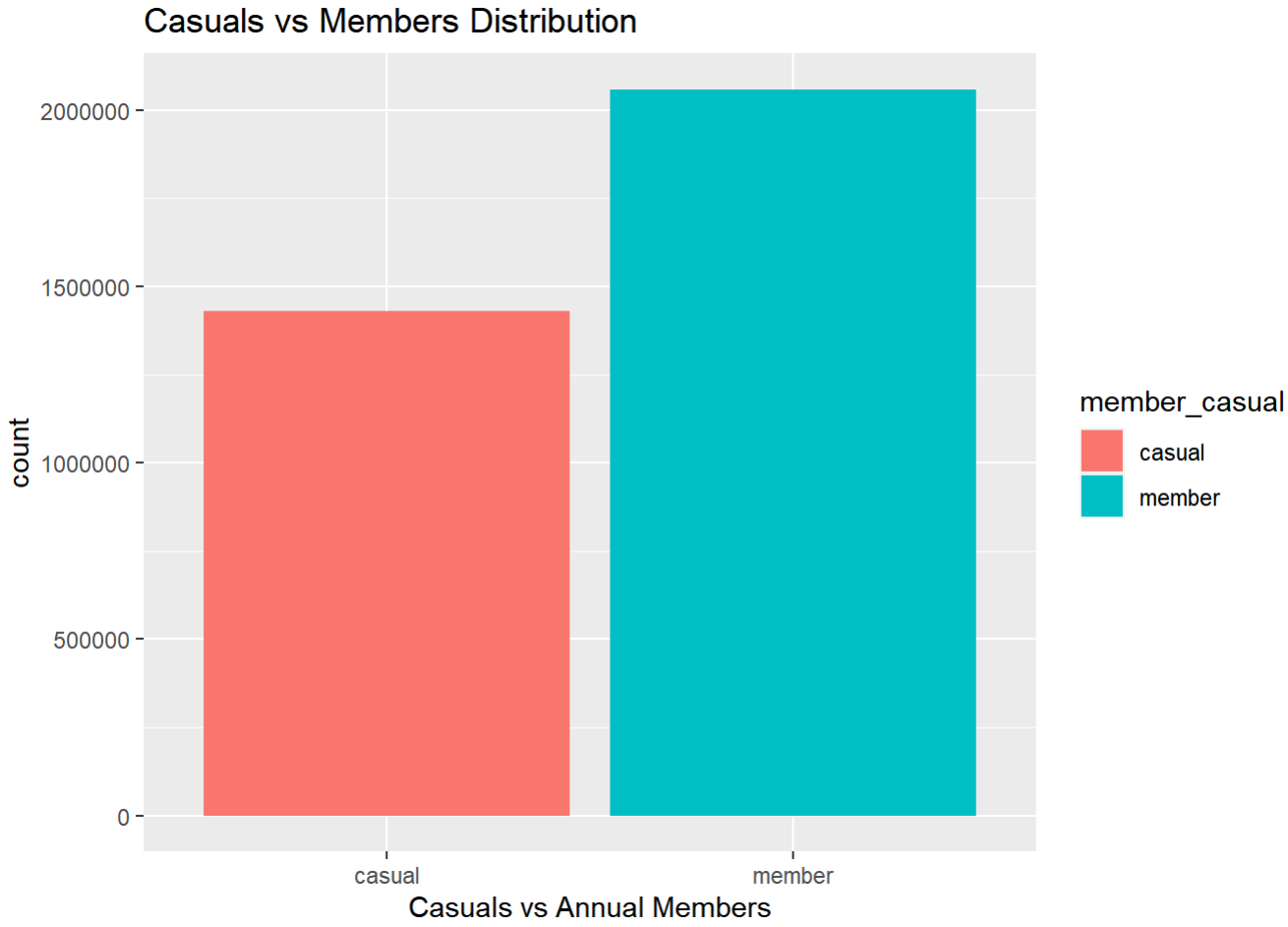
Finding number of casuals vs number of annual members

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

```
## # A tibble: 2 x 3
##   member_casual   count Percentage
##   <chr>           <int>      <dbl>
## 1 casual        1430351       41.0
## 2 member        2059188       59.0
```

Plotting number of casuals vs annuals

```
ggplot(rides, aes(member_casual, fill=member_casual)) + geom_bar() +
  labs(x="Casuals vs Annual Members", title="Casuals vs Members Distribution")
```

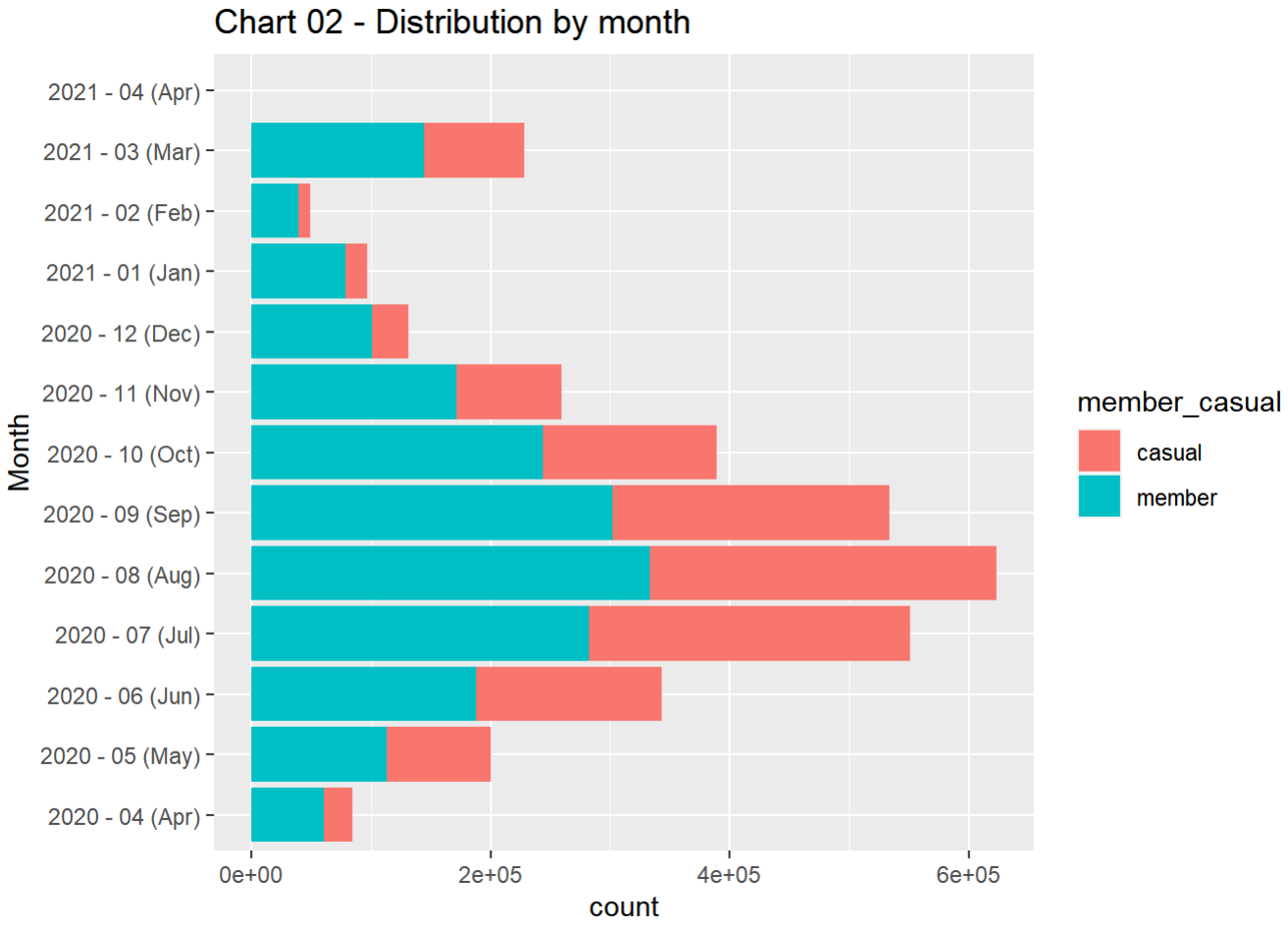


Monthly Distribution of the Data

```
rides %>%
  group_by(year_month) %>%
  summarise(count = length(ride_id),
            'Percentage' = (length(ride_id) / nrow(rides)) * 100,
            'members_p' = (sum(member_casual == "member") / length(ride_id)) * 100,
            'casual_p' = (sum(member_casual == "casual") / length(ride_id)) * 100,
            '% Difference' = members_p - casual_p)
```

```
## # A tibble: 13 x 6
##    year_month       count Percentage members_p casual_p `% Difference`
##    <chr>            <int>      <dbl>     <dbl>    <dbl>          <dbl>
##  1 2020 - 04 (Apr)  84744    2.43         72.1     27.9           44.3
##  2 2020 - 05 (May) 200306    5.74         56.6     43.4           13.2
##  3 2020 - 06 (Jun) 342714    9.82         54.9     45.1            9.83
##  4 2020 - 07 (Jul) 550894   15.8          51.2     48.8            2.38
##  5 2020 - 08 (Aug) 623009   17.9          53.4     46.6            6.86
##  6 2020 - 09 (Sep) 532975   15.3          56.7     43.3           13.4
##  7 2020 - 10 (Oct) 388865   11.1          62.7     37.3           25.4
##  8 2020 - 11 (Nov) 259716    7.44         66.1     33.9           32.2
##  9 2020 - 12 (Dec) 131364    3.76         77.1     22.9           54.2
## 10 2021 - 01 (Jan)  96834    2.77         81.3     18.7           62.6
## 11 2021 - 02 (Feb)  49622    1.42         79.6     20.4           59.2
## 12 2021 - 03 (Mar) 228420    6.55         63.2     36.8           26.4
## 13 2021 - 04 (Apr)     76    0.00218      63.2     36.8           26.3
```

```
rides %>%
  ggplot(aes(year_month, fill=member_casual)) +
  geom_bar() +
  labs(x="Month", title="Chart 02 - Distribution by month") +
  coord_flip()
```



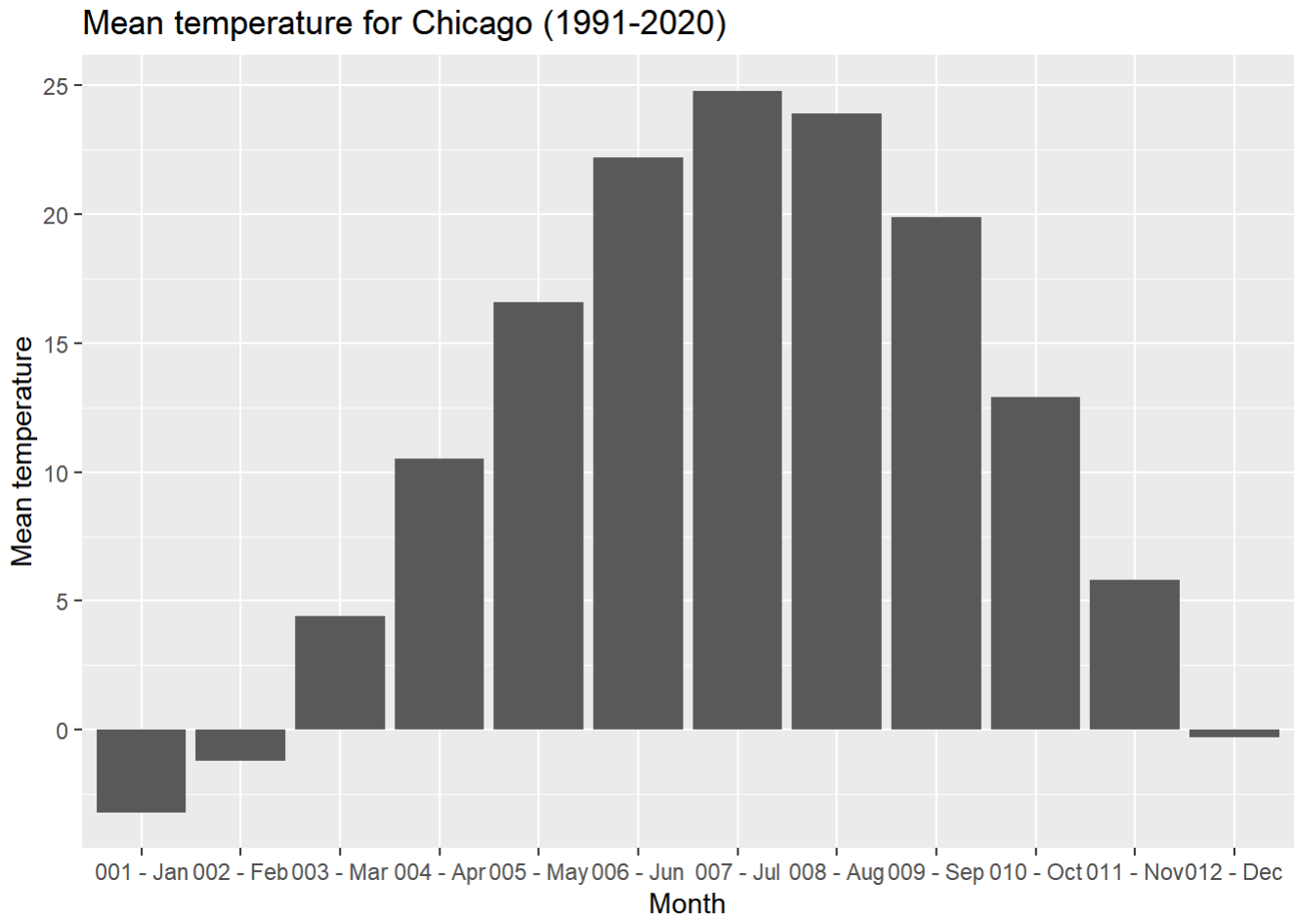
Insights:

1. There are more riders in 2020.
2. August has the biggest number of data points(18% of the dataset).
3. We have more members than riders in every month.

Let us observe how the temperature affects number of riders

```
chicago_mean_temp <- c(-3.2, -1.2, 4.4, 10.5, 16.6, 22.2, 24.8, 23.9, 19.9, 12.9, 5.8, -0.3)
month <- c("001 - Jan","002 - Feb","003 - Mar","004 - Apr","005 - May","006 - Jun","007 - Jul",
           "008 - Aug","009 - Sep","010 - Oct","011 - Nov","012 - Dec")
```

```
data.frame(month, chicago_mean_temp) %>%
  ggplot(aes(x=month, y=chicago_mean_temp)) +
  labs(x="Month", y="Mean temperature", title="Mean temperature for Chicago (1991-2020)") + geom
_col()
```



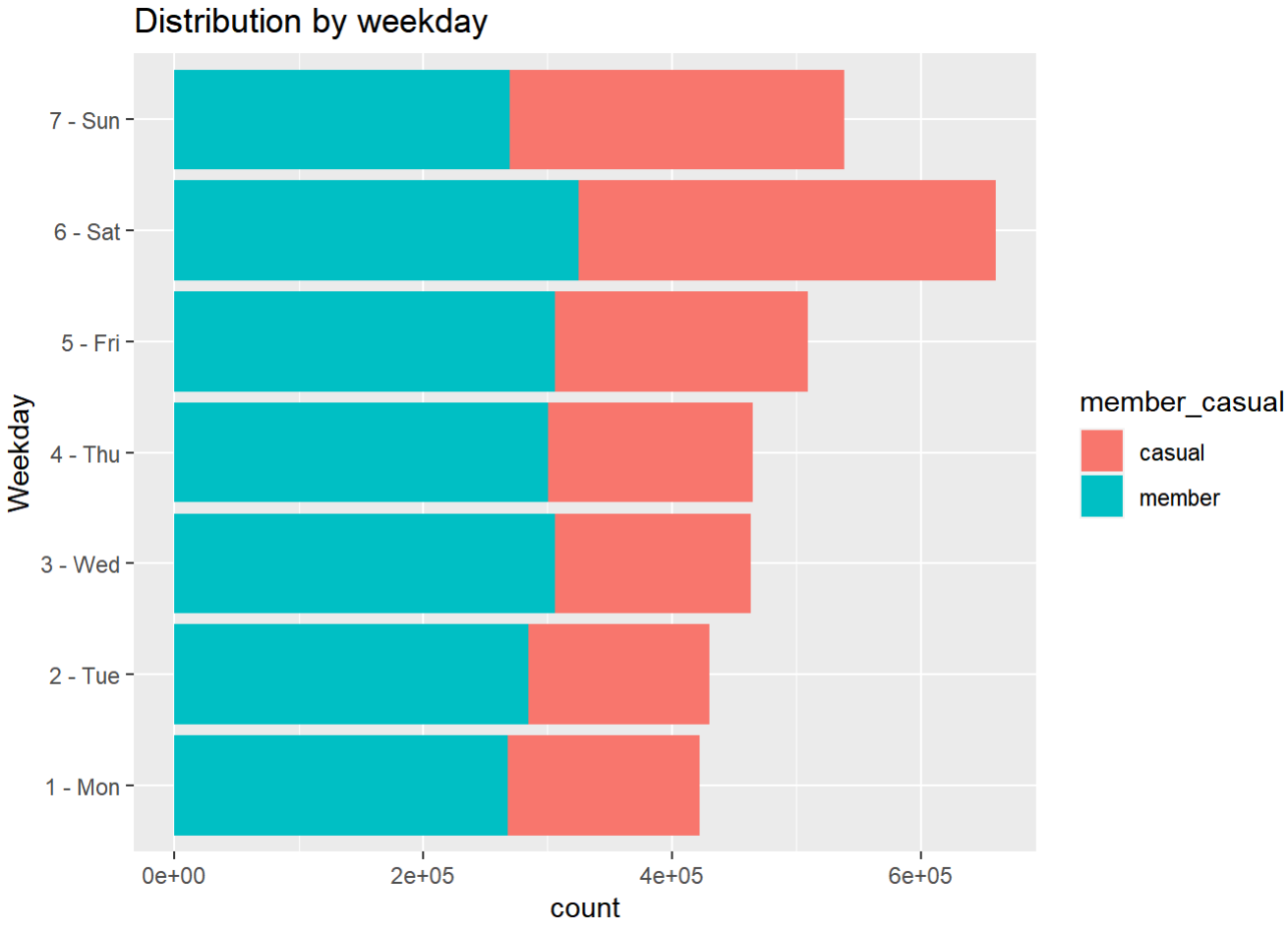
1. As we can see the temperature highly affects the number of riders in every month
2. The colder the weather, the lesser the bike riders

Week Day Distribution

```
rides %>%
  group_by(weekday) %>%
  summarise(count = length(ride_id),
            '%' = (length(ride_id) / nrow(rides)) * 100,
            'members_p' = (sum(member_casual == "member") / length(ride_id)) * 100,
            'casual_p' = (sum(member_casual == "casual") / length(ride_id)) * 100,
            '% Difference' = members_p - casual_p)
```

```
## # A tibble: 7 x 6
##   weekday  count   `%` members_p casual_p `% Difference`
##   <chr>    <int> <dbl>     <dbl>    <dbl>          <dbl>
## 1 1 - Mon 421946  12.1      63.6     36.4       27.2
## 2 2 - Tue 430569  12.3      66.1     33.9       32.3
## 3 3 - Wed 463746  13.3      65.9     34.1       31.8
## 4 4 - Thu 465302  13.3      64.6     35.4       29.1
## 5 5 - Fri 509224  14.6      60.1     39.9       20.1
## 6 6 - Sat 660251  18.9      49.2     50.8       -1.58
## 7 7 - Sun 538501  15.4      50.0     50.0        0.00241
```

```
ggplot(rides, aes(weekday, fill=member_casual)) +
  geom_bar() +
  labs(x="Weekday", title="Distribution by weekday") +
  coord_flip()
```



Insights:

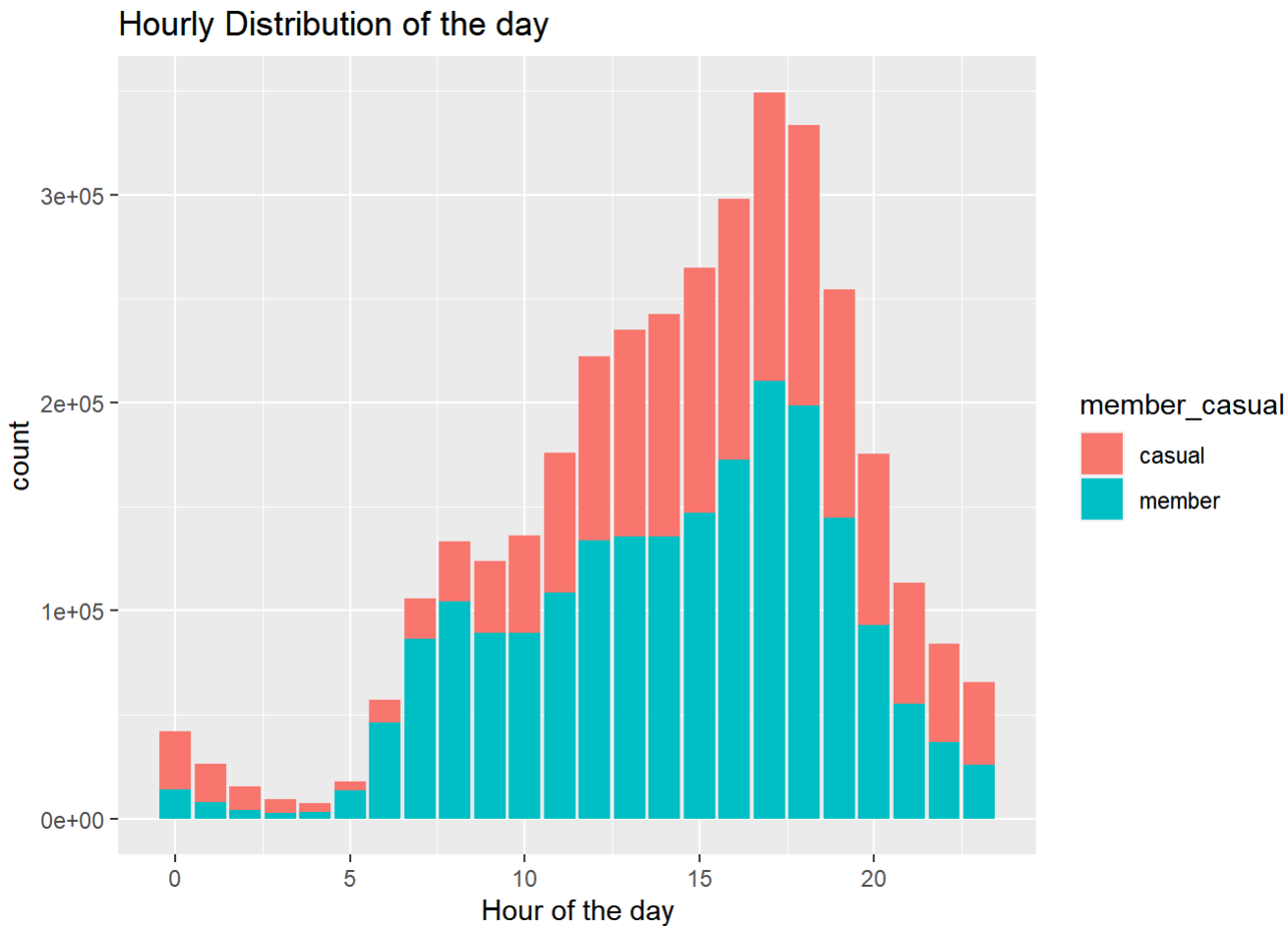
1. The highest volume of riders is on weekends.
2. Saturday being the highest.
3. More casual riders are seen on weekends.

Hourly Distribution

```
rides %>%
  group_by(hour) %>%
  summarise(count = length(ride_id),
            '%' = (length(ride_id) / nrow(rides)) * 100,
            'members_p' = (sum(member_casual == "member") / length(ride_id)) * 100,
            'casual_p' = (sum(member_casual == "casual") / length(ride_id)) * 100,
            '% Difference' = members_p - casual_p)
```

```
## # A tibble: 24 x 6
##     hour  count   `%` members_p casual_p `% Difference`
##    <int>  <int> <dbl>     <dbl>    <dbl>          <dbl>
##  1     0  41924 1.20       33.4     66.6          -33.2
##  2     1  26372 0.756      29.5     70.5          -41.1
##  3     2  15386 0.441      27.5     72.5          -45.1
##  4     3   9038 0.259      27.6     72.4          -44.7
##  5     4   7391 0.212      41.3     58.7          -17.5
##  6     5  17987 0.515      75.0     25.0           50.0
##  7     6  56915 1.63       81.5     18.5           63.0
##  8     7 106045 3.04       81.6     18.4           63.2
##  9     8 133253 3.82       78.4     21.6           56.8
## 10     9 123699 3.54       72.0     28.0           44.0
## # ... with 14 more rows
```

```
rides %>%
  ggplot(aes(hour, fill=member_casual)) +
  labs(x="Hour of the day", title="Hourly Distribution of the day") +geom_bar()
```

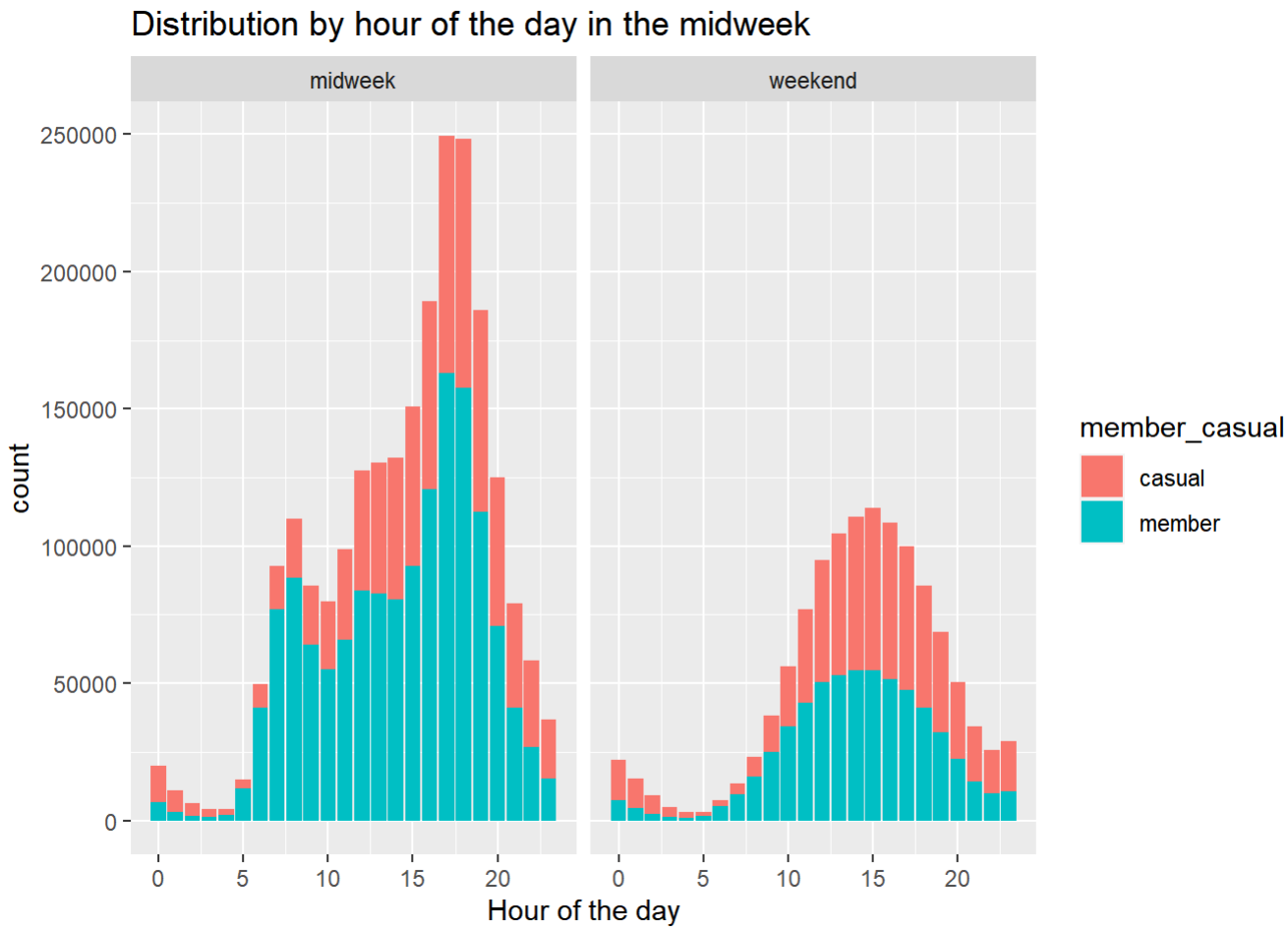


Insights:

1. There are more bikers in the afternoon.
2. There are more members in the morning from 5 am to 11 am.
3. There are more casuals in the afternoon from 11 pm to 4 am.

Weekends vs Mid Week Distribution

```
rides %>%
  mutate(type_of_weekday = ifelse(weekday == '6 - Sat' | weekday == '7 - Sun', 'weekend', 'midwe
ek')) %>%
  ggplot(aes(hour, fill=member_casual)) +
  labs(x="Hour of the day", title="Distribution by hour of the day in the midweek") +
  geom_bar() +
  facet_wrap(~ type_of_weekday)
```



Insights:

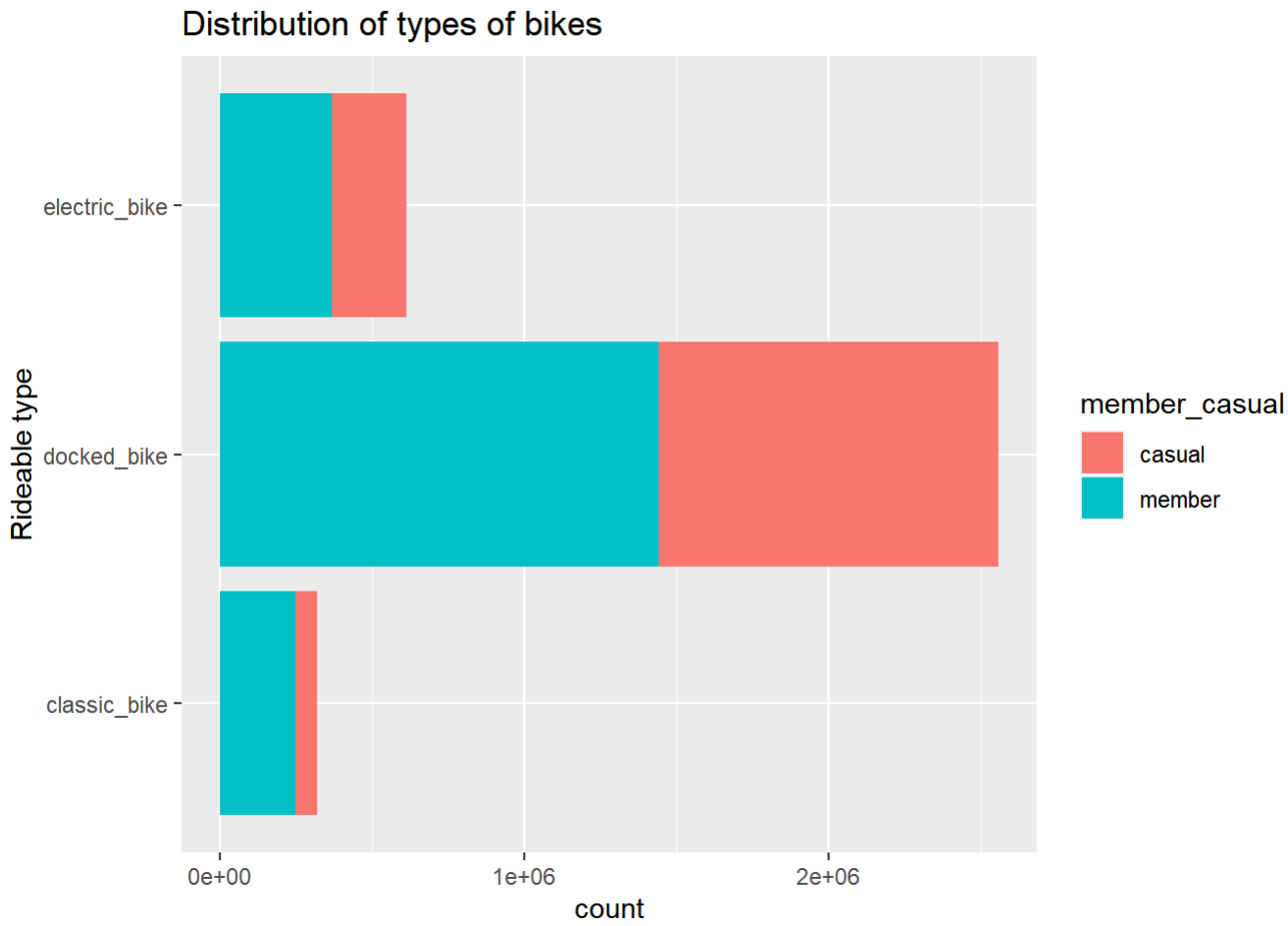
1. Weekends have mre casuals from 11 am to 6 pm.
2. There is huge rise in weekdays from 5 pm to 6 pm, then it drops.
3. On weekends there is smooth flow which rises in the afternoon.

Type of Rideable bikes

```
rides %>%
  group_by(rideable_type) %>%
  summarise(count = length(ride_id),
            '%' = (length(ride_id) / nrow(rides)) * 100,
            'members_p' = (sum(member_casual == "member") / length(ride_id)) * 100,
            'casual_p' = (sum(member_casual == "casual") / length(ride_id)) * 100,
            '% Difference' = members_p - casual_p)
```

```
## # A tibble: 3 x 6
##   rideable_type   count   `%` members_p casual_p `% Difference`
##   <chr>           <int> <dbl>     <dbl>    <dbl>          <dbl>
## 1 classic_bike   319873  9.17      77.9     22.1           55.7
## 2 docked_bike   2558260 73.3       56.4     43.6           12.7
## 3 electric_bike  611406 17.5       60.3     39.7           20.5
```

```
ggplot(rides, aes(rideable_type, fill=member_casual)) +
  labs(x="Rideable type", title="Distribution of types of bikes") + geom_bar() +   coord_flip()
```



Insights:

1. Docked bikes have highest volume of riders.
2. Members prefer the classic and electric bikes.

SHARE PHASE

Key Insights:

1. Members have the biggest proportion of the dataset than casuals.
2. There are more riders at the last semester of 2020. August recorded the biggest count of data points which was 18% of the dataset.
3. Temperature heavily influences the number of rides in each month.
4. Weekends have highest number of riders.There are more riders in the afternoon.

Members vs Casual Riders:

1. Members have highest volume of data, except for saturdays.
2. Weekends have more casuals than members.
3. There are more members in the morning from 5am to 11 am.
4. There are more casuals from 11pm to 4am.
5. There is an increase from 6am to 8am on weekdays for members. Next big rise is from 5pm to 6pm.
6. Members prefer classic bikes.
7. Casuals have more riding time than members.

Conclusion:

1. Bikes are used as an recreational activity on weekends.
2. Temperature affects the number of riders.
3. Members use bikes for daily activities like going to work.

ACT PHASE

Now the marketing teams can use these insights for increasing the number of members and converting casuals to annual members.