*Task*:

**Scenario**

You are 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, your 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  
  
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?

Lily Moreno (Marketing director and you’re manager) has assigned you the first question to answer: How do annual members and casual riders use Cyclistic bikes differently?

You will produce a report with the following deliverables:

1. A clear statement of the business task

2. A description of all data sources used

3. Documentation of any cleaning or manipulation of data

4. A summary of your analysis

5. Supporting visualizations and key findings

6. Your top three recommendations based on your analysis

**The Data**

The data itself is available [here](https://divvy-tripdata.s3.amazonaws.com/index.html). The data is compiled monthly into CSV files, each file contains the same columns which cover a variety of aspects of the bike including bike type, duration, member vs. casual user, and longitude and latitude of the starting and end station among others.

**Data Preparation/Cleaning:**

*R:*

I primarily used R for the data preparation, investigation and cleaning. I chose R as it can handle large amounts of data and is able to give high level overviews of this info with relative ease. I did use it for data visualizations as well, though that was mostly for practice.

*R Code:*  
Used to pull in packages that I will use regularly.

library(tidyverse)

library(skimr)

library(lubridate)

library(janitor)

library(scales)

library(mapview)

Combined all files into one data frame and then investigated the data frame itself. Also created a local version of the combined file for ease of use.

trips <-

list.files(path = "./",

pattern = "\*tripdata.csv",

full.names = T) %>%

map\_df(~read\_csv(., col\_types = cols(.default = "c")))

write\_csv(trips,"./trips.csv”, row.names=FALSE)

glimpse(trips)

trip = trips %>%

remove\_empty(which = c("cols", "rows")) %>%

clean\_names()

Cleaned the names and removed empty columns

trips\_1 = trip %>%

mutate(

start\_lat = as.numeric(start\_lat),

start\_lng = as.numeric(start\_lng),

end\_lat = as.numeric(end\_lat),

end\_lng = as.numeric(end\_lng)

)

glimpse(trips\_1)

Changed latitudinal and longitudinal data into numbers. Reviewed data

colSums(is.na(trips\_1))

Reviewed NA values, as the missing values represented less than 1% of each column, this seemed to not need any further cleaning/review.

Formatted Date and Time Columns and create columns based on month, day, week, weekday, hour started at and trip time.

trips\_2 = trips\_1 %>%

mutate(

started\_at = ymd\_hms(as\_datetime(started\_at)),

ended\_at = ymd\_hms(as\_datetime(ended\_at))

)

glimpse(trips\_2)  
  
trips\_3 = trips\_2 %>%

mutate(

hour\_start = hour(started\_at),

weekday = wday(started\_at, label = T, abbr = F),

month = month(started\_at, label = T, abbr =F),

day = day(started\_at),

week = strftime(started\_at, format = "%V"),

trip\_time = difftime(ended\_at, started\_at, units = "mins")

)

glimpse(trips\_3)

Renamed fields and columns for simplification and checked for missing values.

trips\_4 = trips\_3 %>%

mutate(

rideable\_type = recode(as\_factor(rideable\_type),

"classic\_bike" = "classic",

"electric\_bike"= "electric",

"docked\_bike" = "docked"),

member\_casual = as\_factor(member\_casual)

)

trips\_4 = trips\_4 %>%

rename(

bikes = rideable\_type,

users = member\_casual

)

colSums(is.na(trips\_4))

Filtered out trip times greater than 1 minute and longer than 24 hours.

trips\_5 = trips\_4 %>%

filter(

between(trip\_time, 1, 1440)

)

Create a statistical overview of the data\*see data analysis below.

trips\_5 %>%

skim\_without\_charts()

Create a data frame with time and check for blank data

trips\_time\_df = trips\_5 %>%

drop\_na(

end\_lat, end\_lng

) %>%

select(

ride\_id, users, bikes, hour\_start, weekday, month, day, week, trip\_time

)

colSums(is.na(trips\_time\_df))

Create a data frame with location and check for blank data

trips\_location\_df = trips\_5 %>%

select(

ride\_id, start\_station\_name, end\_station\_name, start\_lat, start\_lng,

end\_lat, end\_lng, users, trip\_time

) %>%

drop\_na(

start\_station\_name, end\_station\_name

)

colSums(is.na(trips\_location\_df))

Write these as csv files for ease of access

write\_csv(trips\_location\_df,"./location.csv”, row.names=FALSE)

write\_csv(trips\_time\_df,"./time.csv”, row.names=FALSE)

*R data Visualization Code*

Split the data into time and location to make visualization easier and then used R to produce some charts.

Created a theme for the charts  
newtheme <- theme\_light() +

theme(plot.title = element\_text(color = "#002949", face = 'bold', size =12),

plot.subtitle = element\_text(color = "#890000", size = 10),

plot.caption = element\_text(color = '#890000', face = 'italic', size =8),

panel.border = element\_rect(color = "#002949", size = 1),

legend.position = "right",

legend.text = element\_text(colour="blue", size=10, face="bold"),

legend.title = element\_text(colour="blue", size=10, face="bold"),

#legend.position='none',

axis.title.x = element\_text(colour = "#890000"),

axis.title.y = element\_text(colour = "#002949"),

axis.text.x = element\_text(angle = 45, hjust = 1, color = '#890000'),

axis.text.y = element\_text(angle = 45, hjust = 1, color = '#002949'),

axis.line = element\_line(color = "#002949", size =1),

)

theme\_set(newtheme)

Created a Trips per Hour and trip duration per hour Chart

ride\_hours %>%

ggplot(aes(hour\_start, nr\_rides, fill = users))+

geom\_col(position = "dodge")+

scale\_y\_continuous()+

labs(

title = "Number of Trips per Hour",

subtitle = "Number of trips for every hour and by users",

x = "hour of the day",

y = "number of rides",

)+

theme()

ride\_hours %>%

ggplot(aes(hour\_start, total\_trip, fill = users))+

geom\_col(show.legend = TRUE, position = "dodge")+

scale\_y\_continuous()+

labs(

title = "Total trip Duration per Hour",

subtitle = "Total duration for every hour segmented by users",

x = "hour of the day",

y = "total duration",

)+

theme()

Made new dataset grouped by weekday

ride\_week = trips\_time\_df %>%

group\_by(

users, weekday

) %>%

summarise(

nr\_rides\_week = n(),

avg\_rides\_week = mean(trip\_time),

total\_duration\_week = sum(trip\_time)

)

Created a trip time by Weekday and a Trip duration by Weekday

ride\_week %>%

ggplot(aes(weekday, nr\_rides\_week, fill = users))+

geom\_col(position = "dodge")+

scale\_y\_continuous(labels = comma)+

labs(

title = "Trips Time by Week Days and Segmented by Users",

subtitle = "Number of trips for every week of the year"

x = "day of the week",

y = "number of trips"

)+

theme()

Grouped by Month

ride\_month = trips\_time\_df %>%

group\_by(

users, month

) %>%

summarise(

nr\_rides\_month = n(),

avg\_rides\_month = mean(trip\_time),

total\_time\_month = sum(trip\_time)

)

Visualized trips by Month and total trip time by Month

ride\_month %>%

ggplot(aes(month, nr\_rides\_month, fill = users))+

geom\_col(position = "dodge")+

scale\_y\_continuous(labels = comma)+

labs(

title = "Number of Trips by Month and Segmented by Users",

subtitle = "Number Trips Time for every Month",

x = "month",

y = " number of trips"

)+

theme()

ride\_month %>%

ggplot(aes(month, total\_time\_month, fill = users))+

geom\_col(position = "dodge")+

scale\_y\_continuous(labels = comma)+

labs(

title = "Total Trips Time by Month and Segmented by Users",

subtitle = "Total Trips Time for every Month",

caption = "Fig 8",

x = "month",

y = "total trips time"

)+

theme()

Segment out popular start and end stations and write these as csv files for further use.

pop\_start\_station = trips\_location\_df %>%

group\_by(

users, start\_station\_name, start\_lat, start\_lng

) %>%

summarise(

nr\_rides\_start = n()

) %>%

arrange(-nr\_rides\_start)

write\_csv(pop\_start\_station,"./ pop\_start\_station.csv”, row.names=FALSE)

pop\_end\_station = trips\_location\_df %>%

group\_by(

users, end\_station\_name, end\_lat, end\_lng

) %>%

summarise(

nr\_rides\_end = n()

) %>%

arrange(-nr\_rides\_end)

write\_csv(pop\_end\_station,"./ pop\_end\_station.csv”, row.names=FALSE)

Visualize Top 10 start and Top 10 End stations

pop\_start\_station[1:10, ] %>%

ggplot(aes(start\_station\_name, nr\_rides\_start, fill = users))+

geom\_col(position = "dodge")+

coord\_flip()+

labs(

title = "Most Popular Start Stations",

subtitle = "Top 10 most popular start stations",

x = "station name",

y = "number of trips"

)+

theme()  
  
pop\_end\_station[1:10,] %>%

ggplot(aes(end\_station\_name, nr\_rides\_end, fill = users))+

geom\_col(position = "dodge")+

coord\_flip()+

labs(

title = "Most Popular End Stations Segmented by Users",

subtitle = "Top 10 most popular end stations"

x = "station name",

y = "number of trips"

)+

theme()

*SQL*

Used to Create top 10 popular End Stations for Tableau

SELECT end\_station\_name,   end\_lat, end\_lng, SUM(nr\_rides\_end) as Total\_Rides

FROM `argon-computer-343415.Start\_Station.End\_Station`

GROUP BY end\_station\_name, end\_lat, end\_lng, nr\_rides\_end

ORDER BY nr\_rides\_end desc

LIMIT 10

Used to Create top 10 popular Start Stations for Tableau

SELECT start\_station\_name,start\_lat, start\_lng, SUM(nr\_rides\_start) as total\_rides

FROM `argon-computer-343415.Start\_Station.Start\_station`

GROUP BY start\_station\_name, start\_lat, start\_lng, nr\_rides\_start

ORDER BY nr\_rides\_start desc

LIMIT 10   
  
**Summary of Analysis:**

* Members represent 3,263,968 (55.76%) whereas Casual users represent 2,589,495 (44.24%) of users captured
* Both Members and Casual users seem to favor weekends (particular Saturday and Sunday), late afternoon and evenings
* The summer months appear to be peak user access with July for Casual (435, 147 rides) and September (385,786) for Members.
* Casual users seem to favor tourist spots along the lake front(Navy Pier in particular) whereas Members focus on locations more in the financial/business district (the loop).

**Data Visualization:**

*Tableau*I used Tableau to create two dashboards: One showing an overview of the data by user type, bike type, and frequency of use. And one showing the location of the ten most popular start and end stations on a map.

[Link Here](https://public.tableau.com/app/profile/kevin.w8746/viz/GoogleDataAnalyticsCapstone_16518570607700/StationGeoData#1)

*R*

Chart, bar chart, histogram

Description automatically generated

Chart, bar chart

Description automatically generated

**Chart, bar chart

Description automatically generated**

**Chart, bar chart

Description automatically generated**

**Chart, bar chart

Description automatically generated**

**Chart, bar chart

Description automatically generated**

**Final Analysis/Recommendations**

* I recommend looking to partner with Navy Pier and other touristy spots in order to boost casual user access
* Additionally, Cyclistic could look to partner with local businesses to over discounts to their employees.
* I believe a weekend pass, summer pass, and afternoon pass may boost sales even more.