Cyclistic Bikeshare Data Analysis: Section Two

Twelve Months of Chicago Ridership

Patrick Holland-Stergar

16 August 2023

Introduction

This is part two of the analysis I conducted on the bikeshare data from the business "Cyclistic". I conducted part 1 using SQL in Google BigQuery, please see part 1 for details. Then, I downloaded the data from BigQuery and ingested it into RStudio, ensuring I also loaded the libraries needed for my analysis.

```
library(data.table) #load data.table package
library(tidyverse) #load tidyverse package
data1 <- fread('C:\\Users\\pholl\\OneDrive\\Desktop\\bikedata.csv')
#load bike user data
```

Question overview

The 5 main questions I worked to answer from the data were:

- How long do customers use the bikes - what is the ride duration?
- How far do the customers ride?
- In what time of the day do the customers use the bikes most?
- What days of the week do the customers use bikes more often?
- What months do the customers use bikes more often?

I will work through these questions one by one, in the same order as I approached them in my visualizations and analysis.

Ridership versus Month

I began by creating a plot for number of rides per customer type per month. My first act was to organize the data of the startMonth column.

```
# I want to plot the number of rides that start in each month
# First, I need to summarize data from the startMonth column

data1$startMonth <-as.factor(data1$startMonth)
```

```
#This converts the startMonth vector to a factor,
# which is Factor is a data structure used
#for fields that takes only a predefined,
# finite number of values (categorical data).
#For example: a data field such as marital
# status may contain only values from single,
#married, separated, divorced, or widowed.
```

Then:

We needed to aggregate data to calculate the number of rides starting in each month and group them by member type.

```
ride_counts <- data1 %>%
  group_by(startMonth, member_casual) %>%
  summarize(number_of_rides = n())

#here, we have piped in the data from data1
# and then grouped it by which month each row of data
# has for startMonth as well as what type of rider was using it.
# Then, we have used the summarize function to sum how many
# rides (how many rows,
# to be specific) occur in each month. This is
# all now put in a new data frame called ride_counts.
```

I discovered an issue when I first tried to create my plot, which was that there was one observation that had it's value for the member_casual column as "member_casual", so I returned to my script and added a filter to eliminate that erroneous data point from my chart. I also filtered out any observations that lacked a value for startMonth.

```
# Filter out rows with missing startMonth values and erroneous member_casual value
ride_counts_filtered <- ride_counts %>%
  filter(!is.na(startMonth),
         member_casual !="member_casual")
```

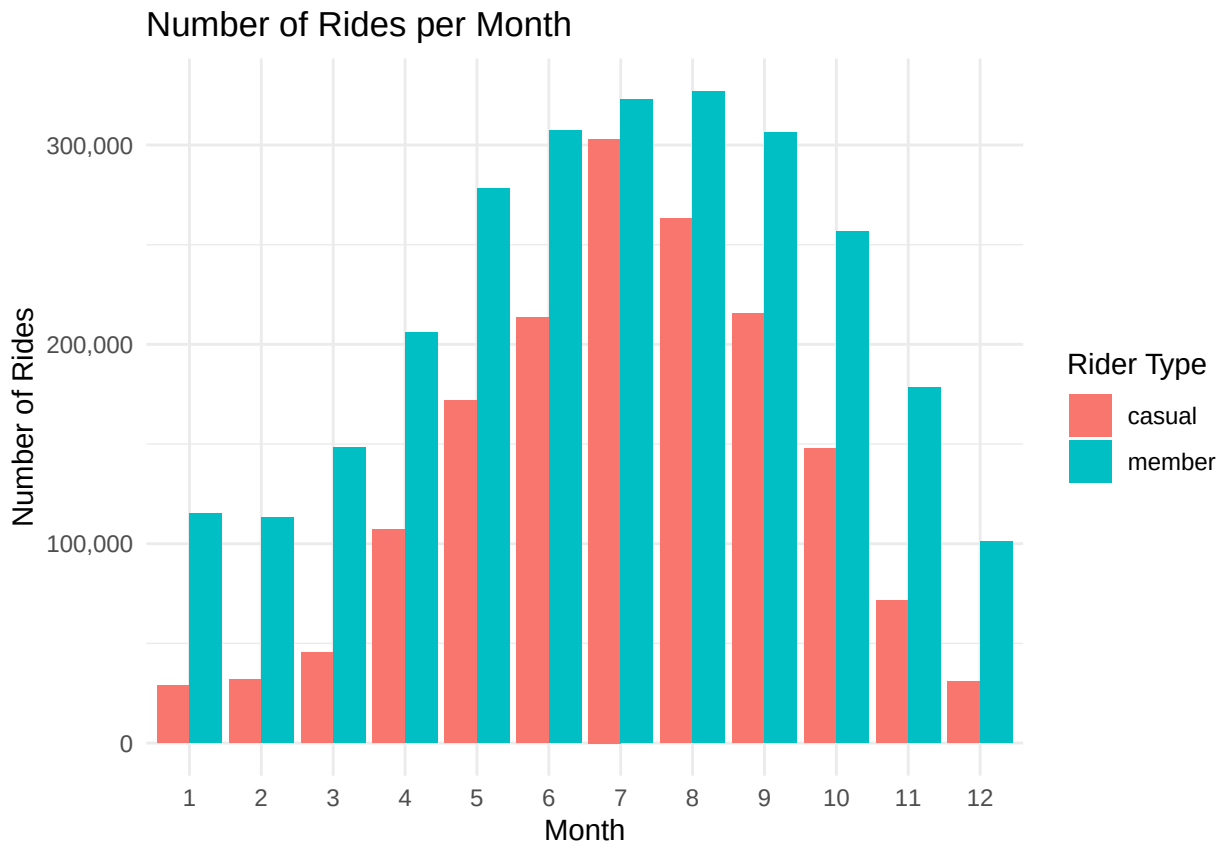
When I first created a plot, I also had the issue that on the x-axis, values of months, which should be ordered from 1 (January) through 12 (December) in numeric order, were not appearing in the correct order in the plot, therefore I manually ordered them correctly with this script:

```
ride_counts_filtered$startMonth <-
  factor(ride_counts_filtered$startMonth,
         levels = as.character(1:12))
```

With the data now organized and filtered, I could create my plot.

```
# Create the bar plot
data_viz <- ggplot(data = ride_counts_filtered,
           aes(x = startMonth,
           y = number_of_rides,
           fill=member_casual)) +
  geom_bar(stat = "identity",
           position = "dodge") +
```

```r
  labs(x = "Month", y = "Number of Rides",
       title = "Number of Rides per Month",
       fill = "Rider Type") +
  theme_minimal() +
  scale_y_continuous(labels = scales::comma)
# Modify y-axis labels to use commas for thousands separator

#Here, we choose the data frame as
# ride_counts_filtered, and plotted the startMonth
# (independent variable) against number of rides
# (dependent variable)
# We also chose to use different fill colors for
# members vs casual riders. Then we specified the plot
# to be a bar chart, and used "identity" to state
# that the row height should be directly dependent
# on the magnitude of rides taken and used "dodge"
# to put the two bars (one per rider type) next to
# each other
# for each month. To ensure the y-axis labels
# are in standard notation, we used the
# scale_y_continuous statement and also
# used the theme_minimal() to specify
# a clean, minimalist theme

print(data_viz)
```



On the x-axis of this chart, the numbers represent months where January = 1 and December = 12.

From this chart, we can gain a few major insights into the behavior of Cyclistic bike riders.

First, we see that throughout the year, more users are members than are casual riders. The discrepancy is greatest in the winter months such as months 1, 2, 3, 11, and 12 on the chart, which correspond to January (1), February (2), March (3), November (11), and December (12), whereas the gap is the smallest in the summer months, but for every single month there are more rides taken by members than by casual riders.

Second we can see that ridership is highest in the summer months and lowest in the winter months, with total ridership in July being roughly 625,000 compared to only about 130,000 in December, meaning Cyclistic bikes are used nearly 5x in the peak of summer compared to the nadir in the winter.

Ridership versus Day of the Week

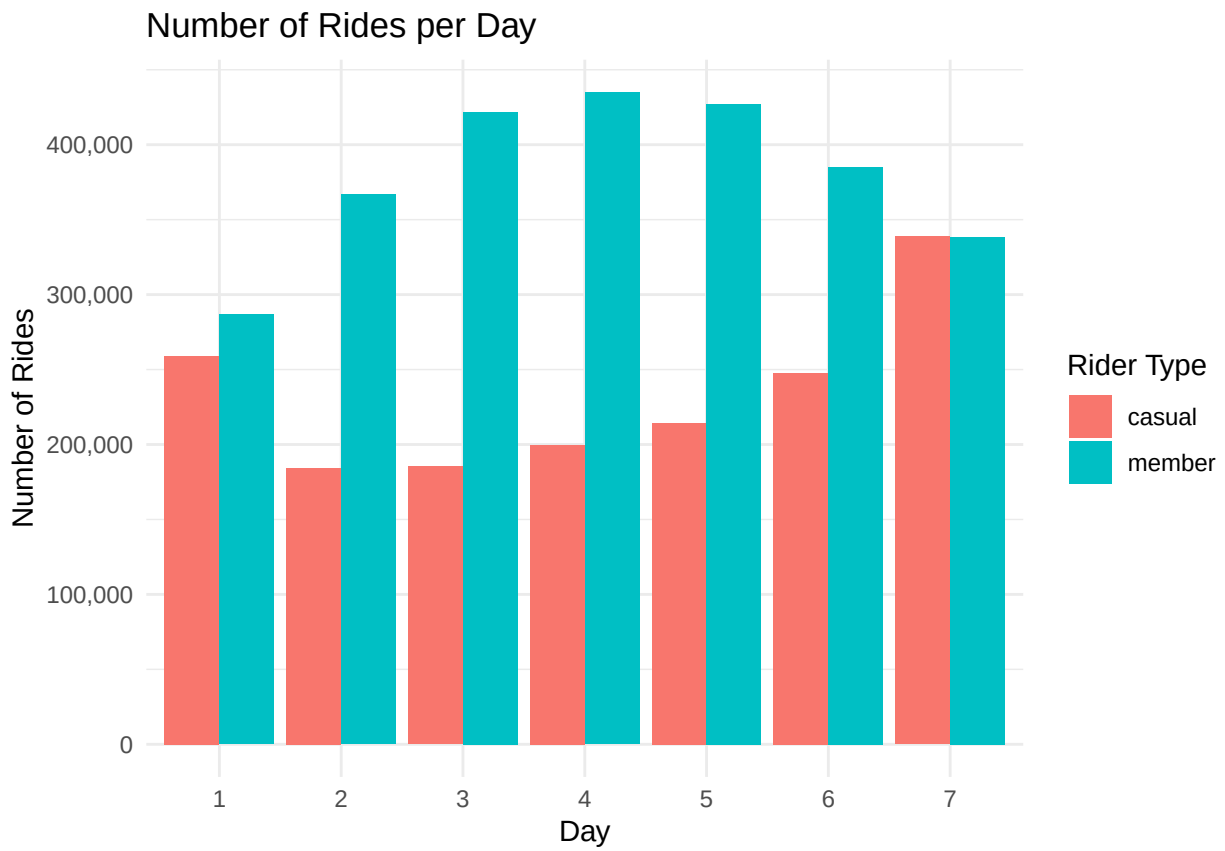
Analyzing ridership on a month by month basis provided insights into the seasonality of Cyclistic's business. Next I wanted to see on a weekly basis what the data showed for ridership. In addition to examining ridership on a day by day basis, I also examined ridership on weekend days versus week days to determine if there were any discernible trends that perhaps could be correlated to recreational bike riders versus those using bikes for their job commutes.

First, I cleaned and organized the relevant data in a fashion similar to that employed and described for the creation of plot #1 (Ridership versus month).

```
data1$startDay <-as.factor(data1$startDay)
#This converts the startDay vector to a factor

#Now, we needed to Aggregate data to calculate the number of
# rides starting in each month and separated by member type

ride_counts2 <- data1 %>%
  group_by(startDay, member_casual) %>%
  summarize(number_of_rides = n())

#here, we have piped in the data from data1
# and then grouped it by which month each ride
# began in as well as which type of rider. Then,
# we have used the summarize function to sum how
# many rides occur on each day of the week. This
# is all now put in a new data frame called
# ride_counts2

# Filter out rows with missing startDay values
# and those not matching a member_casual value of "member" or
#"casual"
ride_counts_filtered2 <- ride_counts2 %>%
  filter(!is.na(startDay),
       member_casual !="member_casual")

# Manually set the order of levels for the startDay
# factor
ride_counts_filtered2$startDay <-
  factor(ride_counts_filtered2$startDay,
       levels = as.character(1:7))
#This is so the plot shows the months in correct
# order from Sunday (1) through Saturday (7)
```

Next, with the data now organized, I created the plot:

```
# Create the bar plot
data_viz <- ggplot(data = ride_counts_filtered2,
       aes(x = startDay, y = number_of_rides,
           fill=member_casual)) +
  geom_bar(stat = "identity",
           position = "dodge") +
  labs(x = "Day", y = "Number of Rides",
       title = "Number of Rides per Day",
       fill = "Rider Type") +
  theme_minimal() +
  scale_y_continuous(labels = scales::comma)


#Here, we choose the data frame as
# ride_counts_filtered2, and plotted the
# startDay (independent variable) against
# number of rides (dependent variable)
# We also chose to use different fill colors
# for members vs casual riders. Then we specified
# the plot to be a bar chart, and used "identity"
# to state
# that the row height should be directly dependent
# on the magnitude of rides taken and used "dodge"
# to put the two bars (one per rider type) next
# to each other
# for each month. To ensure the y-axis labels are
# in standard notation, we used the scale_y_continuous
# statement and also used the theme_minimal() to specify
# a clean, minimalist theme

print(data_viz)
```



On this plot, the numbers represent days of week in order beginning with Sunday, so Sunday = 1 and Saturday = 7.

From this plot, we can see that ridership for casual users dips during the week but reaches its peak during the weekend, whereas the trend for membership rides is the opposite - it crests during the middle of the week but reaches its lowest points on the weekends.

It could be surmised from this trend that casual riders are more likely to be recreational riders, people using the bikes on a night out, or other users who are not using the bikes to commute to their workplaces or using them for work-related purposes (e.g. delivery people), and the opposite could be true for members - they are more likely to use the bikes to travel to a workplace, from a workplace, or use the bike during their work.

I plotted ridership on weekend days versus that on week days to further investigate this issue.

First, I created a new column to classify each ride as "weekday" or "weekend" depending on the day when the ride started.

```
data1 <- data1 %>%
  mutate(isweekend = ifelse(startDay %in% c("1", "7"), TRUE, FALSE))
#Here, I used the mutate function to create a new column
# "isweekend" whose value depended on whether the ride
# began on a weekend day, which is 1 or 7 in the startDay column,
# or not.

#Now, we need to Aggregate data to calculate the number of rides
# starting in each month and separated by member type

ride_counts4 <- data1 %>%
```

```
  group_by(member_casual, isweekend) %>%
  summarize(number_of_rides = n())


#here, we have piped in the data from data1 and then grouped it
# by which month each row of data
# has for isweekend as well as what type of rider was using it.
# Then, we have
# used the summarize function to sum how many rides occur per
# type of day.
# This is all now put in a new data frame called ride_counts4
```

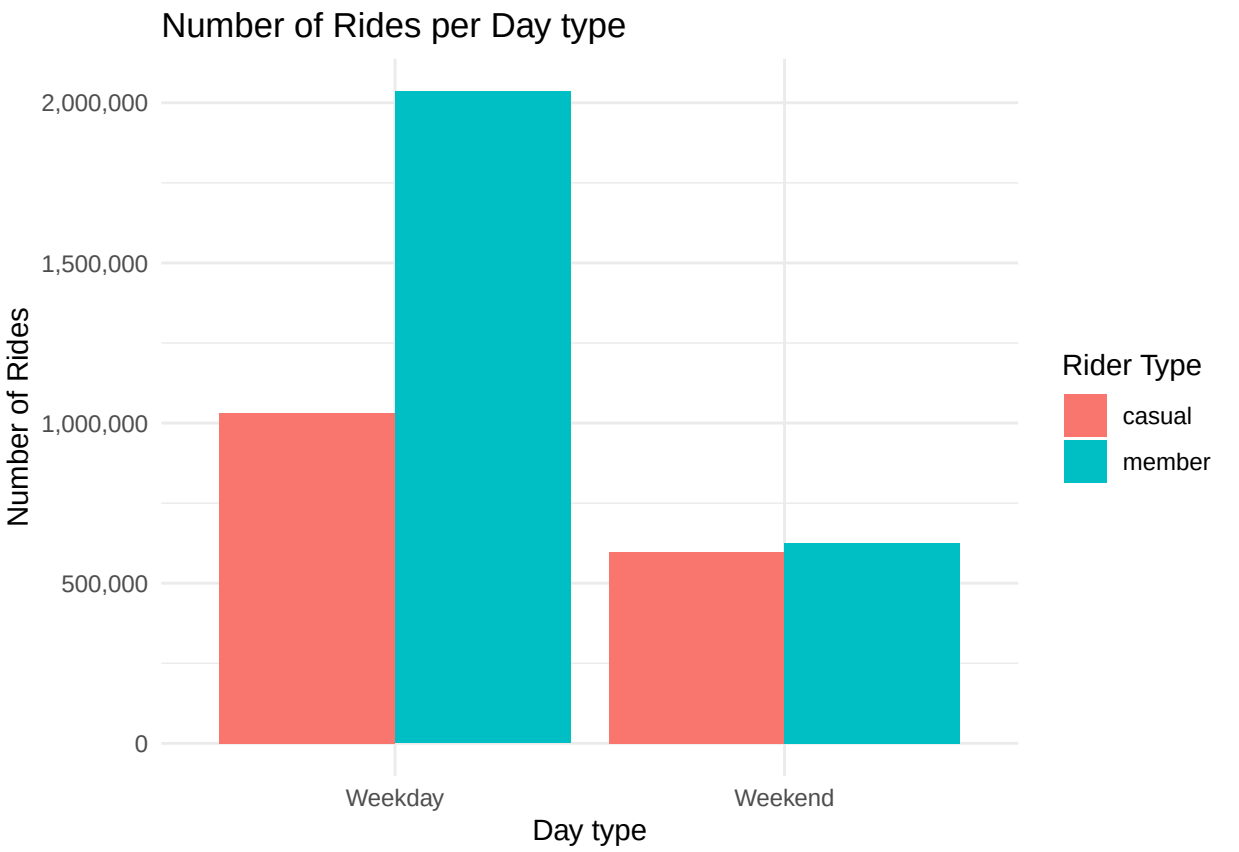
Then, I ensured my data was properly filtered and organized in advance of plotting it.

```
ride_counts_filtered4 <- ride_counts4 %>%
  filter(!is.na(isweekend),
       member_casual !="member_casual")
#I needed to filter out any missing values from isweekend
# and ensure that only those observations where member_casual
# is equal to member or casual were included
```

Finally, I created the plot.

```
# Create the bar plot
data_viz <- ggplot(data = ride_counts_filtered4, aes(x = isweekend,
y = number_of_rides,
fill=member_casual)) +
geom_bar(stat = "identity",
position = "dodge") +
labs(x = "Day type", y = "Number of Rides",
title = "Number of Rides per Day type",
fill = "Rider Type") +
theme_minimal() +
scale_y_continuous(labels = scales::comma) +
scale_x_discrete(labels = c("Weekday", "Weekend"))



# Modify y-axis labels to use commas for thousands separator
#Here, we choose the data frame as ride_counts_filtered4,
# and plotted isweekend (independent variable) against
# number of rides (dependent variable)
# We also chose to use different fill colors for members
# vs casual riders. Then we specified the plot to be a bar
# chart, and used "identity" to state
# that the row height should be directly dependent on the
# magnitude of rides taken and used "dodge" to put the
# two bars (one per rider type) next to each other
# for each month. To ensure the y-axis labels are in
# standard notation, we used the scale_y_continuous
# statement and also used the theme_minimal() to specify
# a clean, minimalist theme
```

```
print(data_viz)
```



This chart clearly illustrates that while for weekends, the ratio of member to casual riders is about 1:1, that ratio increases to about 2:1 for weekdays. In other words, it is much more likely that on a weekday a rider will be a member and much more common for weekend riders to be a casual rider.

All together, the data supports the idea that under current conditions, members tend to use the bikes in a different manner than casual riders.

Ridership versus Hour of Use

The third important comparison for time versus ridership is considering the time of day when the bikes are ridden - is there any difference between when casual riders use the bikes compared to when members use them?

First, I cleaned and organized the data to ready it for plotting.

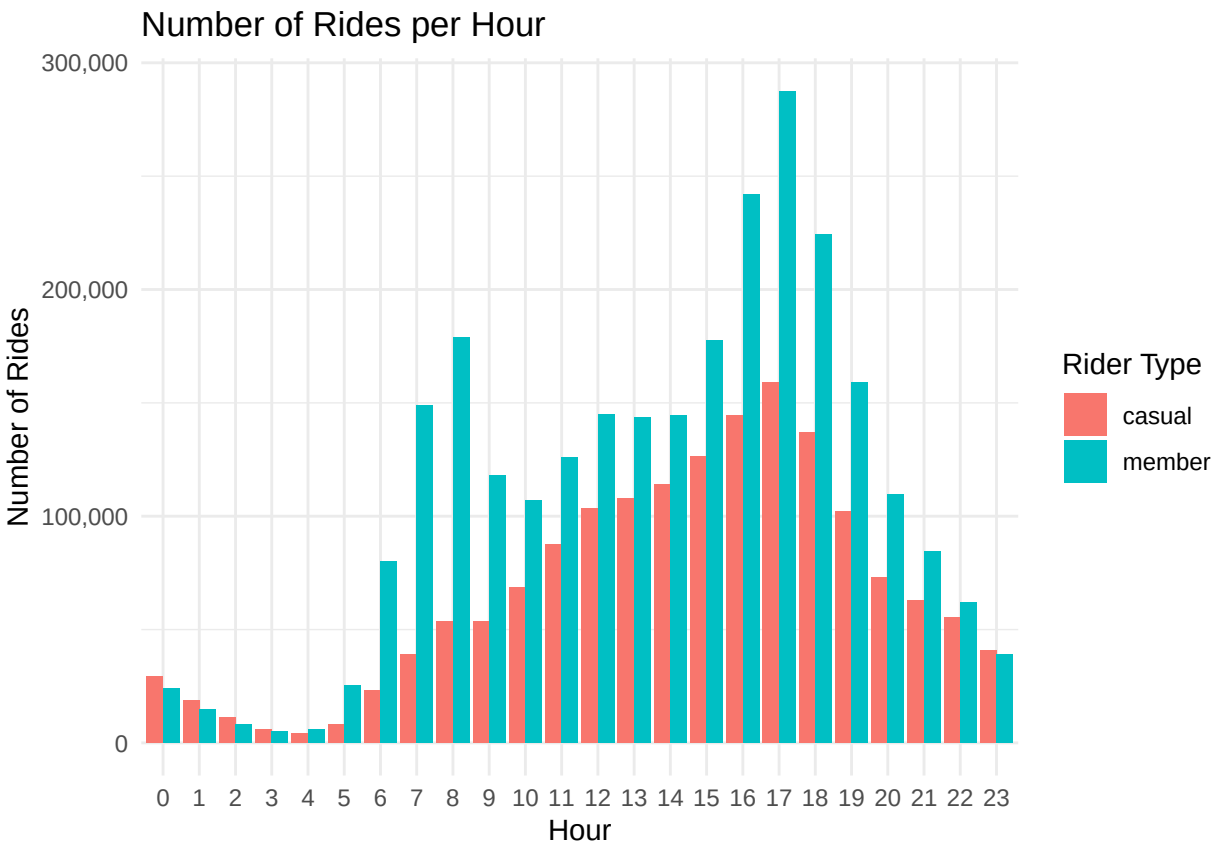
```
#Aggregate data to calculate the number of rides
# starting at each hour of the day and group them
# by member type

ride_counts_hour <- data1 %>%
  group_by(startHour, member_casual) %>%
  summarize(number_of_rides = n())
```

```r
#here, we have piped in the data from data1 and then
# grouped it by which month each row of data
# has for startHour as well as what type of rider
# was using it. Then, we have used the summarize
# function to sum how many rides started in each
# hour in the day. This is all now put in a new data
# frame called ride_counts_hour.


# Filter out rows with missing startHour values
ride_counts_hour_filtered <- ride_counts_hour %>%
  filter(!is.na(startHour),
     member_casual !="member_casual")
#I filtered the data so that no missing values were
# in the startHour column and so that only the two valid
#values for member_casual were present


# Manually set the order of levels for the startMonth factor
ride_counts_hour_filtered$startHour <-
  factor(ride_counts_hour_filtered$startHour,
       levels = as.character(0:23))
#This is so the plot shows the months in correct order
# from Midnight (0) through Eleven pm (23)
```

Then, I plotted the data.

```r
# Create the bar plot
data_viz <- ggplot(data = ride_counts_hour_filtered,
       aes(x = startHour,
       y = number_of_rides,
     fill=member_casual)) +
  geom_bar(stat = "identity",
       position = "dodge") +
  labs(x = "Hour", y = "Number of Rides",
     title = "Number of Rides per Hour",
     fill = "Rider Type") +
  theme_minimal() +
  scale_y_continuous(labels = scales::comma)

# Modify y-axis labels to use commas for
# thousands separator
#Here, we choose the data frame as
# ride_counts_hour_filtered, and plotted the
# startHour (independent variable) against
# number of rides (dependent variable)
# We also chose to use different fill colors
# for members vs casual riders. Then we specified
# the plot to be a bar chart, and used "identity" to state
# that the row height should be directly dependent
# on the magnitude of rides taken and used "dodge"
# to put the two bars (one per rider type) next to each other
# for each month. To ensure the y-axis labels are
```

```
# in standard notation, we used the scale_y_continuous
# statement and also used the theme_minimal() to specify
# a clean, minimalist theme

print(data_viz)
```



From this chart, we can see that membership use of the bikes spikes in the morning hours between 7 and 9 am and then again around the hours of 3 pm to 7 pm, which would be consistent with usage by commuters. In contrast, the pattern of use by casual members shows a generally consistent increase from 4 am to 5pm and then a generally consistent decrease between 5 pm and 4 am. It is notable that while at 5 pm the ratio of member rides to casual rides is about 1.75:1, by 11 pm it decreases to roughly 1:1 and then from midnight through 3 am there are more casual riders using the bikes than members. This could correlate to casual users being more likely than members to use bikes while they are enjoying nightlife activities.

Ride Duration

We've examined the relationship between how many rides are taken relative to time (hour of the day, day of the week, month of the year), but we have not yet examined if there is any difference between how long rides by members are in comparison rides by casual users. To examine this issue, I plotted ride duration versus by month, grouped by rider type.

I began by organizing and filtered the data before plotting it.

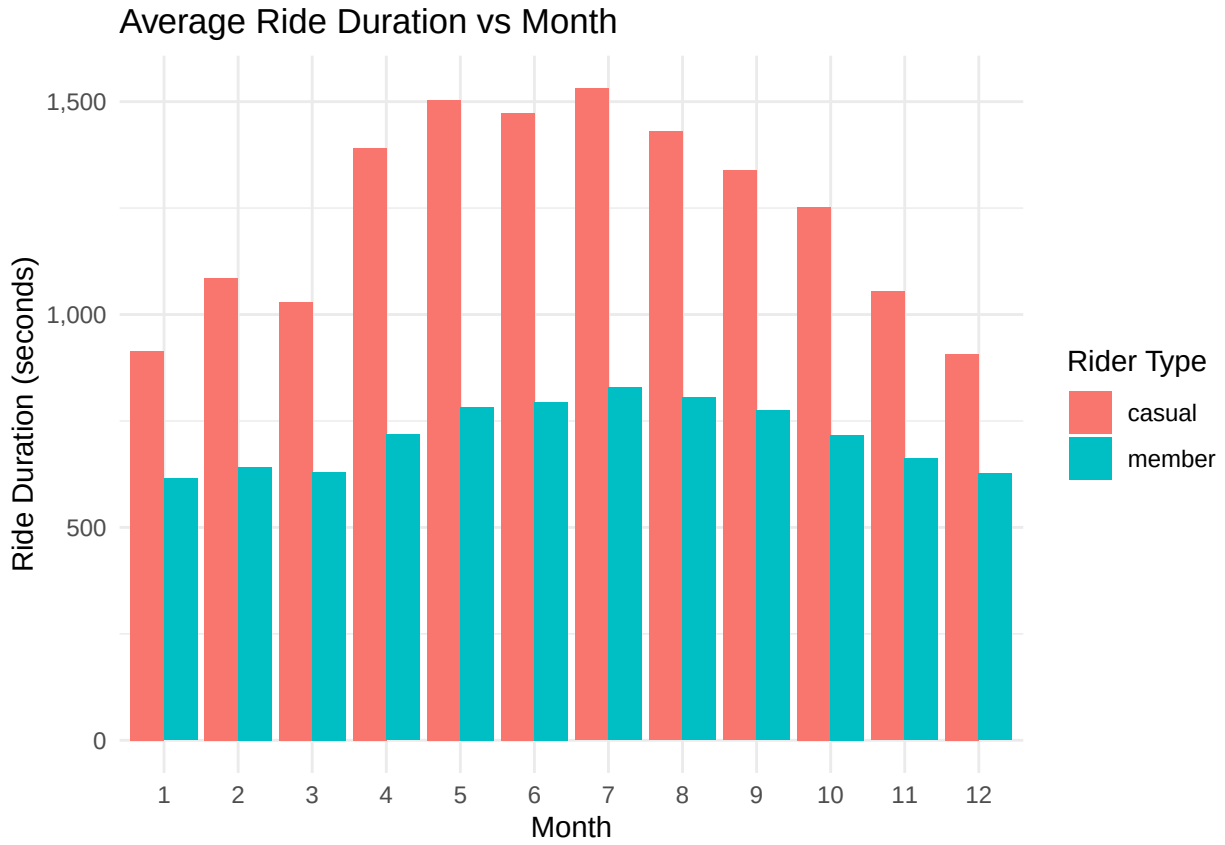
```
data1 <- data1 %>%
  mutate(ride_duration = as.integer(gsub("[^0-9]", "", ride_duration)))
```

```
# I needed to convert the column ride_duration
# from chr to integer


#Now, we need to Aggregate data to calculate the
# number of rides starting in each month and
# separated by member type

ride_duration_tbl <- data1 %>%
  group_by(startMonth, member_casual) %>%
  summarize(avg_duration = mean(ride_duration,
            na.rm = TRUE))



#here, we have piped in the data from data1
# and then grouped it by rider type and by month.
# Then, we have used the summarize function to
# determine the average duration in seconds for
# rides by month.

# Next, I filtered out rows with missing startMonth
# values and any that lacked a valid value for member_casual
ride_duration_tbl_flt <- ride_duration_tbl %>%
  filter(!is.na(startMonth),
     member_casual !="member_casual")


# Manually set the order of levels for the startMonth factor
ride_duration_tbl_flt$startMonth <-
  factor(ride_duration_tbl_flt$startMonth,
         levels = as.character(1:12))
#This is so the plot shows the months in correct order
# from January (1) through December (12)
```

The data now ready, I plotted average ride duration by rider type against month.

```
# Create the bar plot
data_viz <- ggplot(data = ride_duration_tbl_flt,
         aes(x = startMonth, y = avg_duration,
         fill=member_casual)) +
  geom_bar(stat = "identity",
       position = "dodge") +
  labs(x = "Month", y = "Ride Duration (seconds)",
       title = "Average Ride Duration vs Month",
       fill = "Rider Type") +
  theme_minimal() +
  scale_y_continuous(labels = scales::comma)

# Modify y-axis labels to use commas for thousands separator
#Here, we choose the data frame as
# ride_duration_tbl_flt, and plotted the
# startMonth (independent variable) against
```

```
# the average duration of rides in that month
# We also chose to use different fill colors
# for members vs casual riders. Then we
# specified the plot to be a bar chart,
# and used "identity" to state
# that the row height should be directly
# dependent on the magnitude of rides taken
# and used "dodge" to put the two bars
# (one per rider type) next to each other
# for each month. To ensure the y-axis
# labels are in standard notation, we
# used the scale_y_continuous statement
# and also used the theme_minimal() to specify
# a clean, minimalist theme

print(data_viz)
```



The chart shows that there is a much smaller range across the span of year in average ride duration for members as compared to casual riders. The lowest values for both occur in the winter months, but while for members the average duration goes only from ~625 seconds in December and January to ~800 seconds in July, the average duration for casual riders increases from ~900 seconds in December and January to ~1550 seconds in July. Therefore, there is more seasonal variability for casual riders than for members. This supports our extant interpretations that members tend to use bikes for activities they must complete regardless of season or weather (e.g. commuting to work) whereas casual riders may use the bikes more for purely voluntary or recreational uses such as scenic bike rides in the summer. Furthermore, it may be possible that casual riders are more likely to not live in Chicago, but we would have to have more data about

the individual users to confirm this and at the very least examine trends on visitors to Chicago to see if in fact more out-of-town people are present in the city in the summer months.

It is also notable that the average duration for a ride is considerably longer for casual riders than for members, regardless of month. The difference ranges from roughly 3 or 4 minutes in December and January to roughly 12 minutes in July. There are a variety of possible interpretations for this difference. They are all speculation without further study, but include the idea that casual riders are using the bikes for less-frequent, longer rides such as scenic bike rides along the shore of Lake Michigan, whereas members may use them for more consistent, "targeted" rides such as the commute from their home to their place of work.

One way to further investigate this point is to examine the average ride distance. This approach is limited by the fact that the available data set only contains information about the duration, starting point, and ending point for each ride. If someone were to return a bike to the same place they began their ride, the available data shows that ride as having a distance of 0.0.

This is not only a limitation because in itself does not provide useful information about how the bicycle was used - e.g. was it used for a quick trip to the grocery store and back, or a longer recreational ride -but also without knowing the speed of the bicycle during the ride and if it was paused for any length of time, the data cannot tell us if a ride that began and ended at the same point but lasted 20 minutes was because the rider actually traveled in a large loop away from the point of origin or if instead they traveled only one block, stopped at a restaurant, bar, park, or other place, forgot to pause the ride on their app, stayed at the place for 15 minutes, and then returned the bike to the point of origin.

Nevertheless, it will be useful to compare average ride distance between the two user types.

To begin, I organized and filtered the data to prepare it for plotting.

```
data1 <- data1 %>%
    mutate(Distance_travelled2 = as.numeric(Distance_travelled2))
```

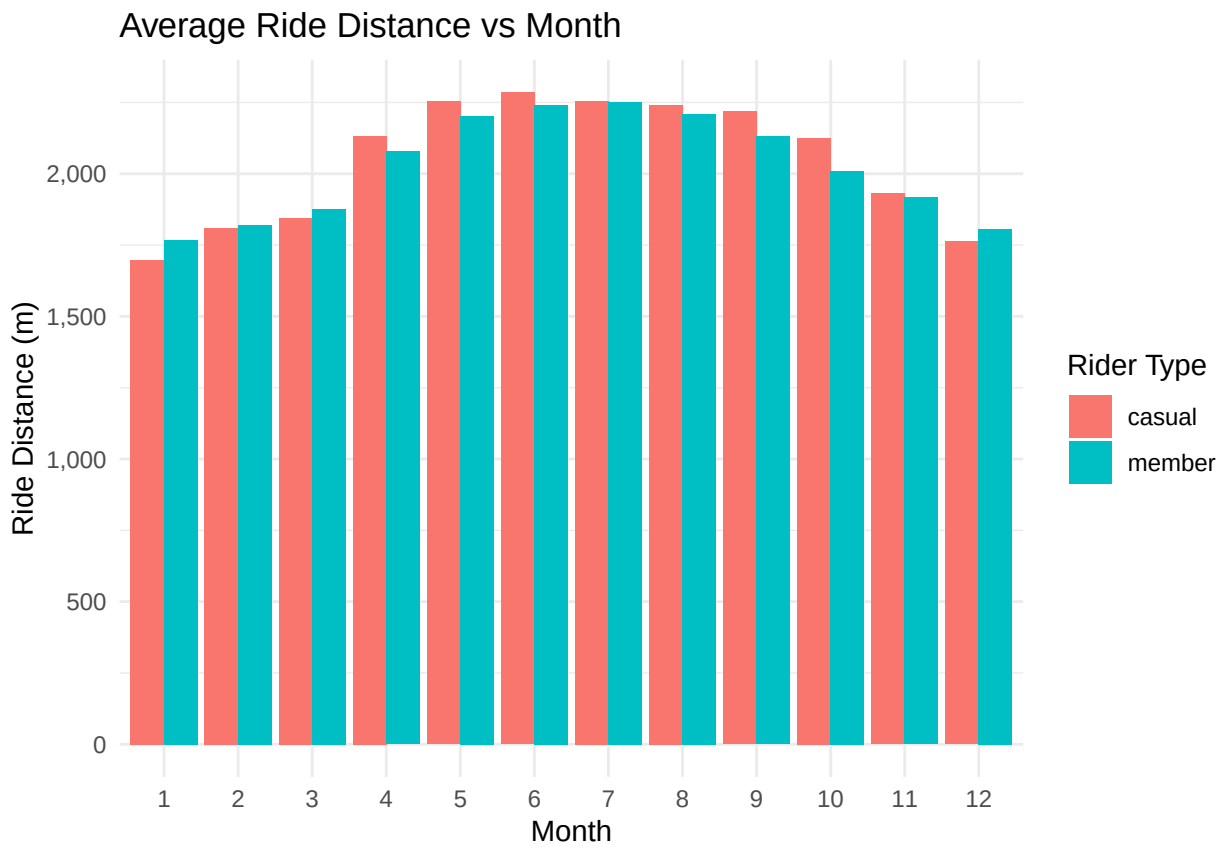
```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `Distance_travelled2 = as.numeric(Distance_travelled2)`.
## Caused by warning:
## ! NAs introduced by coercion
```

```
# I needed to convert the column Distance_travelled2 from chr to a number data
# type



# Now, we need to Aggregate data to calculate the average distance for rides in
# each month grouped by member type

avg_distance_tbl <- data1 %>%
    group_by(startMonth, member_casual) %>%
    summarize(avg_distance = (sum((Distance_travelled2), na.rm = TRUE)/n()))
# this bottom line takes the sum of all the distance travelled per month and
# divides it by the total number of bike rides taken in that month


# Filter out rows with missing startMonth values and ensure all member_casual
# observations are not an erroneous value
avg_distance_tbl_flt <- avg_distance_tbl %>%
    filter(!is.na(startMonth), member_casual != "member_casual")
```

```r
# Manually set the order of levels for the startMonth factor
avg_distance_tbl_flt$startMonth <- factor(avg_distance_tbl_flt$startMonth, levels = as.character(1:12))
# This is so the plot shows the months in correct order from January (1)
# through December (12)
```

Then, I plotted the data as a bar chart.

```r
# Create the bar plot
data_viz <- ggplot(data = avg_distance_tbl_flt, aes(x = startMonth, y = (avg_distance),
    fill = member_casual)) + geom_bar(stat = "identity", position = "dodge") + labs(x = "Month",
    y = "Ride Distance (m)", title = "Average Ride Distance vs Month", fill = "Rider Type") +
    theme_minimal() + scale_y_continuous(labels = scales::comma)

# Modify y-axis labels to use commas for thousands separator Here, we choose
# the data frame as avg_distance_tbl_flt, and plotted the startMonth
# (independent variable) against average distance for rides in that month
# (dependent variable) We also chose to use different fill colors for members
# vs casual riders. Then we specified the plot to be a bar chart, and used
# 'identity' to state that the row height should be directly dependent on the
# magnitude of rides taken and used 'dodge' to put the two bars (one per rider
# type) next to each other for each month. To ensure the y-axis labels are in
# standard notation, we used the scale_y_continuous statement and also used the
# theme_minimal() to specify a clean, minimalist theme

print(data_viz)
```



From this chart we can see that the average ride distance for both types of riders is very close for every month throughout the year, and though there is a small seasonal variation, the magnitude of difference does not appear to be significant (ranging from roughly 1.75 km in January to about 2.25 km in July). It does not seem that any differences are significant between the two groups.

Insights and Findings

- Throughout the year, more Cyclistic bike riders are members than are casual riders. There were roughly 100,000 more rides taken in the past 12 months by members than casual riders, and member rides outnumber casual rider rides for every single month.
- During the course of a week, on average, ridership for members increases during the workweek and tapers off on the weekend, whereas casual riders use the bikes more often on weekends and less often on weekdays.
- Ridership for casual riders is much more seasonal than for members. While the difference in number of rides between the least busiest and busiest months for members is about 3x (i.e. about 120,000 at a low to 330,000 in the busiest month), the difference for casual riders is closer to 10x (i.e. 35,000 vs 300,000). Ridership is at its highest in the summer months.
- On weekends, the ratio of casual riders to members is about 1:1, while during the week about twice as many riders are members as are casual riders.
- Casual riders, per ride on average, ride for considerably longer than members. This ranges from about 5 minutes longer at a minimum in December to about 12 minutes longer in May and July.
- Ride distance is very similar, throughout the year, for both groups.

The differences in riding behaviours between the casual user group and the member group overall support the thesis that members tend to use the bikes for work purposes while casual riders use the bikes for social and recreational rides. On the weekends, those members who do use the bikes are most likely using them for social and recreational rides just as the casual users are, but during the week, there is a clear distinction in the data between how the bikes are used between the groups, lending support to the idea that members are using the bikes for work purposes Monday through Friday while casual riders most often use the bikes for recreational and social purposes every day of the week. Since outdoor bicycle riding for social and recreational purposes would vary more with seasons of the year, relative to the need to travel to and from a place of work, the fact that the data shows far more seasonality in bicycle use by casual riders as compared to members aligns with the concept of different primary use cases for each rider group.

Recommendations for the Business

1. The temporal focus for a marketing campaign aimed at converting casual riders to members should be in the period May through September. While available data does not contain any information about unique individual riders, casual ridership is highly seasonal. Whereas the number of casual rides is below 100,000 from December through March, it is above 150,000 from May through September (and above 200,000 from June through September). Therefore, the summer months see far more use of Cyclistic's bicycles - presumably due to a mix of casual riders who are active during the winter using the bikes much more and use by casual riders who only use the bicyles during the summer months. Accordingly, an advertising campaign that utilizes channels of advertising messages physically on Cyclistic bicycles and station, via the Cyclistic app, and through other channels will reach a larger number of casual users if conducted from May through September.

2. An advertising campaign aimed at converting casual riders to members should align with the extant behaviour of casual riders and explain how a membership would allow them to continue or expand their current uses of the bikes for recreational or social purposes while providing benefits such as streamlined access or financial savings. For example, advertising images depicting a group of friends using the bikes on a fun, casual ride (i.e. casual clothes, good weather) would be more likely to resonate than imagery showing a rider in a business suit seemingly on their way to their office. Current casual riders are most likely using the bikes more often for social or recreational purposes, therefore advertising strategy should accept and "lean into" this reality as opposed to attempting to convince current casual riders to adopt an entirely new use case for the bicycles.

3. Since casual members use the bikes more heavily on weekends than on weekdays, a new type of membership could be developed that would be specifically tailored to weekend riders. For example, the membership could offer unlimited rides only on Saturdays and Sundays and the ability to reserve bikes on only those days in exchange for a lower membership fee than the standard seven-days-a-week unlimited membership. Another product option would be a seasonal membership valid only during the summer months - this could attract both seasonal riders who live in Chicago full-time as well as anyone who is spending time in Chicago only in the warmer months (e.g. tourists or "snowbirds").

Options for Further Analysis

- Conduct a survey with a statistically significant sample of the casual rider population, asking them their needs, preferences, goals, and current behaviour as it relates to Cyclistic's bicycles. This will provide qualitative data to supplement the quantitative ridership data and supply insights into who exactly they are - tourists, students, working professionals, retirees, etc. - so that more specific marketing approaches can be employed.

- Collect more data about ride costs for casual members, and if possible collect or gain access to data that would make it possible to connect data to each unique rider. This approach would allow microtargeted marketing to each person (such an a dashboard in the Cyclistic app) that could, for example, show the financial savings created by adopting a membership instead of paying for each ride individually.