Cyclistic Case Study | Google Data Analytics Course

Joshua Buxton

2025-01-27

The following data set and business task are part of the Google Data Analytics Certification Case Study. The goal of this case study is to perform my own, independent data analysis in order to showcase the skills and knowledge we have acquired over the course of completing the certification. The following data set is from a fictional cycling company and we, as the junior data analyst, have been tasked with answering the following question: **How do annual members and casual riders use Cyclistic bikes differently?**

Step 1: Ask

As per the course, the first step in our data analysis process is Ask. This means that we need to ask the higher ups who assign us our business task exactly what it is they are looking for. It is important to make sure that the work you complete is what they need. Obviously, since this is a predetermined situation with our question and data already laid out, performing the task is not possible, but if I was given the task in person, the following are some questions I might ask:

- In regards to the differences between members and casual riders, are there any particular metrics that you're interested in?
- Is there a specific time frame from the data that should be focused on?

Step 2: Prepare

This step of the data analysis process involves collecting and organizing the relevant data for your analysis. This part is already complete as the course provided the data set for us. The second part of prepare stage is cleaning / transforming the data. The following are the steps I took in order to prep the data for analysis.

I first wanted to get a look at the data using the `glimpse` function. This allows me to not only view some of the first rows of data for each column, but I can also see the row / column count, column names, and column data types. From this, I was able to determine if all columns had the correct data type and followed the same naming convention. I was also able to begin looking at which variables would be important in answering the business task.

```
glimpse(divvy)
```

```
## Rows: 335,075
## Columns: 13
## $ ride_id            <chr> "578DDD7CE1771FFA", "78B141C50102ABA6", "1E794CF363~
## $ rideable_type      <chr> "classic_bike", "classic_bike", "classic_bike", "cl~
## $ started_at         <dttm> 2024-11-07 19:21:58, 2024-11-22 14:49:00, 2024-11-~
## $ ended_at           <dttm> 2024-11-07 19:28:57, 2024-11-22 14:56:15, 2024-11-~
## $ start_station_name <chr> "Walsh Park", "Walsh Park", "Walsh Park", "Clark St~
## $ start_station_id   <chr> "18067", "18067", "18067", "TA1307000039", "TA13070~
## $ end_station_name   <chr> "Leavitt St & North Ave", "Leavitt St & Armitage Av~
## $ end_station_id     <chr> "TA1308000005", "TA1309000029", "13133", "TA1307000~
## $ start_lat          <dbl> 41.91461, 41.91461, 41.91461, 41.90297, 41.93650, 4~
## $ start_lng          <dbl> -87.66797, -87.66797, -87.66797, -87.63128, -87.647~
## $ end_lat            <dbl> 41.91053, 41.91781, 41.91598, 41.93125, 41.89228, 4~
## $ end_lng            <dbl> -87.68231, -87.68244, -87.67733, -87.64434, -87.612~
## $ member_casual      <chr> "member", "member", "member", "member", "casual", "~
```

```
divvy %>%
  summarise_all(~ sum(is.na(.))) %>%
  pivot_longer(everything(), names_to = "column", values_to = "na_count") %>%
  filter(na_count > 0)
```

```
## # A tibble: 6 x 2
##   column             na_count
##   <chr>                 <int>
## 1 start_station_name    56203
## 2 start_station_id      56203
## 3 end_station_name      57644
## 4 end_station_id        57644
## 5 end_lat                 273
## 6 end_lng                 273
```

The first task I performed was removing the rows that had end times earlier than start times. Clearly, there is an error somewhere in input process and since this only affected a few dozen rows, dropping these rows is negligible. In the above code output, we can see the six columns that contain NA values. In regards to my analysis, only start_station_name / start_station_id are of value. However, dropping these rows would make me lose nearly 17% of the data and could introduce bias when looking at certain variables. For this reason, I will filter the data in future visualizations when necessary.

The next step involved transforming the data. In order to better understand the differences between the two demographics, I created ride length variable the contains the length of the ride in minutes and a day of the week variable which records the starting day of the ride (e.g. Monday).

```
clean_data <- function(data) {

  # Removes rows in which the end time is before the start time (incorrect times)
  data <- data %>%
    filter(ended_at > started_at)

  # Creates a new variable that records the ride length in minutes
  data <- data %>%
    mutate(
      ride_length = as.numeric(round((ended_at - started_at) / 60, 1))
    ) %>%
    filter(ride_length >= 2)

  # Creates a new variable that records the day of the ride (day ride started)
  data <- data %>%
    mutate(
      day_of_week = weekdays(started_at)
    )

  return(data)
}

divvy_clean <- clean_data(divvy)
```

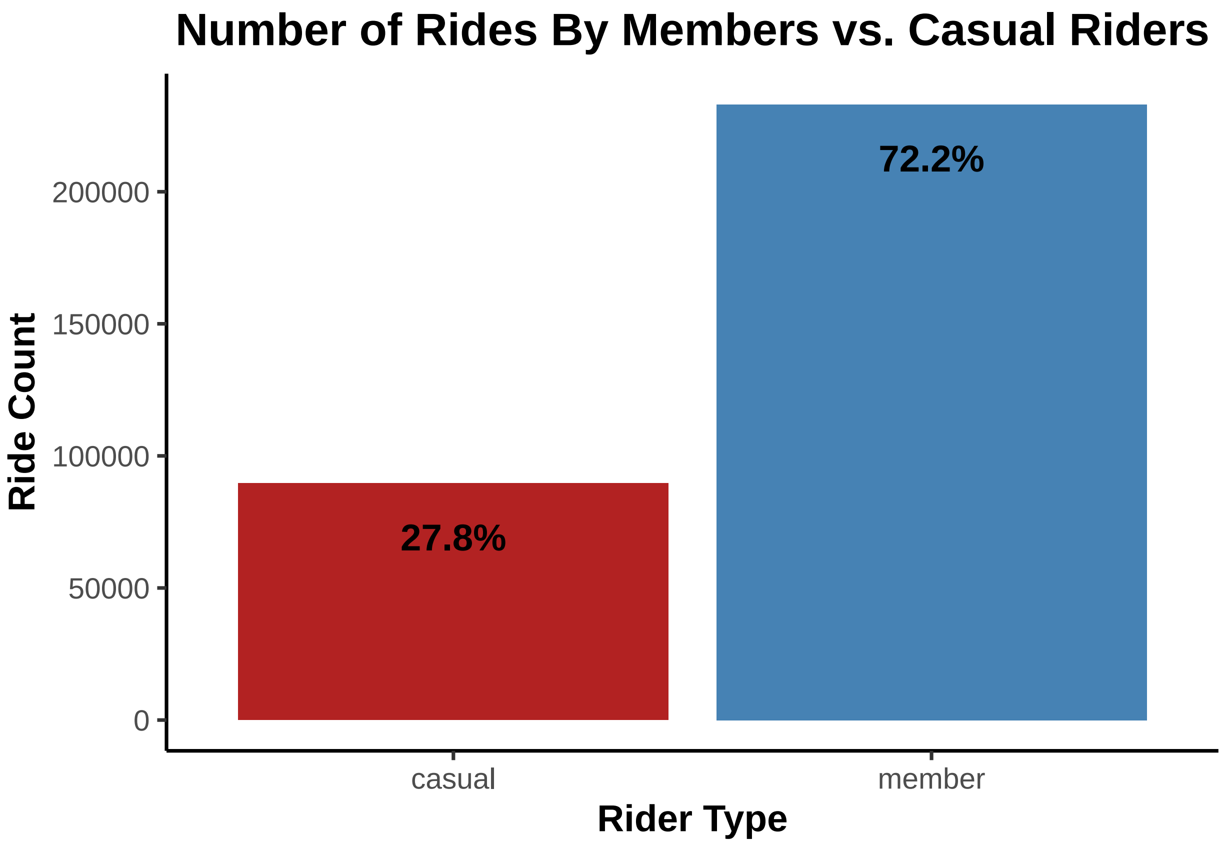
Step 3: Process

The process step of the data analysis involves exploring the data and finding any trends or anomalies through visualizations, summary tables, etc. The first visualization I made is a simple bar chart the displays how many of the rides were members vs casual riders. From this chart, we can see that roughly 75% of the rides were from members while only 25% were casual.

In the plot, the x-axis is represents `member_casual` while the y-axis displays the count of each type. The `geom_text()` displays the percentage of casual vs. member riders on the bars. `scale_fill_manual` applies custom colors and `guides()` removes the legend. Finally, `labs()`, `theme_classic()` and `theme()` add a title and axis labels, swap the theme for a cleaner alternative and adjust the position and formatting of the title / labels.

```r
divvy_clean %>%
  ggplot(aes(x = member_casual)) +
  geom_bar(aes(fill = member_casual)) +
  geom_text(
    aes(label = sprintf("%.1f%%", (..count.. / sum(..count..)) * 100)),
    stat = "count",
    vjust = 2.5,
    fontface = "bold",
    size = 5
  ) +
  scale_fill_manual(
    values = c("member" = "steelblue", "casual" = "firebrick"),
  ) +
  guides(fill = "none") +
  labs(
    title = "Number of Rides By Members vs. Casual Riders",
    x = "Rider Type",
    y = "Ride Count"
  ) +
  theme_classic(base_size = 14) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title = element_text(face = "bold"),
    strip.text = element_text(face = "bold")
  )
```

```
## Warning: The dot-dot notation (`..count..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(count)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

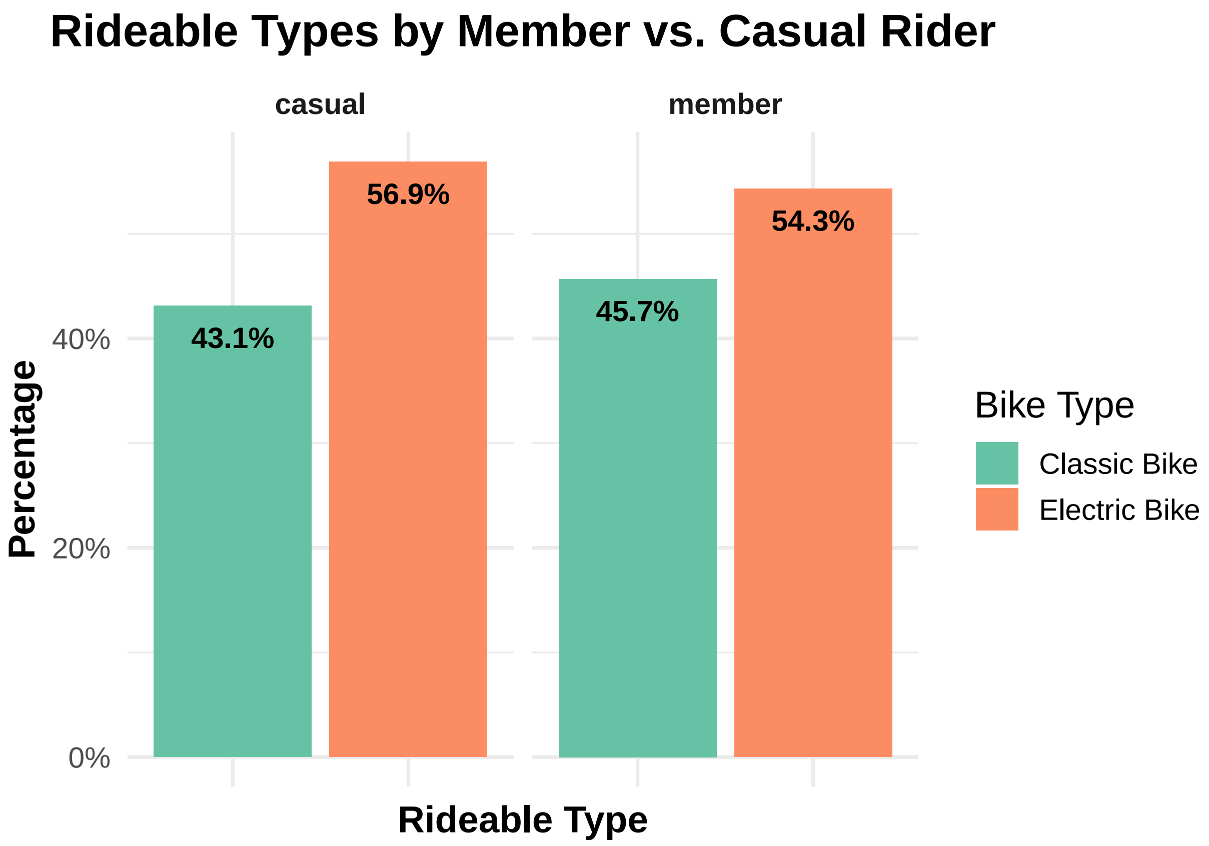


The second graph I created is also a bar chart, but it displays the percentage of classic vs electric bike rides for both casual and member riders. Looking at the graph, we can see that both casual and member riders had very similar percentages for classic vs. electric bike rides.

To create this graph, I first grouped the data by `member_casual` and `rideable_type` to count the number of rides for each category (casual-classic, casual-electric, member-classic, member-electric). Next, I re-grouped the data by `member_casual` to calculate the percentage of each bike type relative to `member_casual`. This allowed me to find the percentage of classic vs. electric bike rides for casual and member riders, relative to their specific counts.

In the plot, the x-axis represents `rideable_type` and the y-axis shows the percentage of each bike type within its member / casual grouping. I used `stat = "identity"` to plot the already calculated percentages from `divvy_percent`. The `facet_wrap(~ member_casual)` created separate plots for member vs. casual riders. Next, `geom_text()` was used to display the percentage labels for each bar while `scale_y_continuous(labels = function(x) paste0(x, "%")` formats the y-axis as percentages. Lastly, `scale_fill_brewer()`, `labs()`, `theme_minimal()` and `theme()` were applied to change the colors, add the title and axis labels, change the plot theme to a cleaner style and position / bold the labels.

```
divvy_percent <- divvy_clean %>%
  group_by(member_casual, rideable_type) %>%
  summarise(count = n(), .groups = "drop") %>%
  group_by(member_casual) %>%
  mutate(percentage = count / sum(count) * 100)

divvy_percent %>%
  ggplot(aes(x = rideable_type, y = percentage, fill = rideable_type)) +
  geom_bar(stat = "identity") +
  facet_wrap(~ member_casual) +
  geom_text(
    aes(label = sprintf("%.1f%%", percentage)),
    vjust = 2,
    fontface = "bold",
    size = 4
  ) +
  scale_y_continuous(labels = function(x) paste0(x, "%")) +
  scale_fill_brewer(
    palette = "Set2",
    labels = c("classic_bike" = "Classic Bike", "electric_bike" = "Electric Bike")
  ) +
  labs(
    title = "Rideable Types by Member vs. Casual Rider",
    x = "Rideable Type",
    y = "Percentage",
    fill = "Bike Type"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title = element_text(face = "bold"),
    strip.text = element_text(face = "bold"),
    axis.text.x = element_blank()
  )
```



The following summary tables show certain metrics for ride lengths according to two categories: casual riders and member riders. These tables consist of the minimum, maximum, mean, median, first quartile, third quartile, interquartile range, standard deviation, skewness and kurtosis values. The key take aways are that casual riders on average, ride for longer than members, the standard deviation from casual riders is higher than member riders meaning there is more spread from the mean in casual riders. Lastly, both groups have extremely high positive skews and kurtosis. The skewness indicates large right tails, which makes sense considering the means for both are in the low tens while they have maxes of roughly 1500. As for kurtosis, the extremely high positive values illustrate similar results, that there is a heavy tail and significant outliers.

```
# Metrics for casual riders
divvy_clean %>%
  filter(member_casual == "casual") %>%
  summarise(
    min = min(ride_length),
    max = max(ride_length),
    mean = mean(ride_length),
    median = median(ride_length),
    q1 = quantile(ride_length, 0.25),
    q3 = quantile(ride_length, 0.75),
    iqr = IQR(ride_length),
    std = sd(ride_length),
    skewness = skewness(ride_length),
    kurtosis = kurtosis(ride_length)
  )
```

```
## # A tibble: 1 x 10
##     min   max  mean median    q1    q3   iqr   std skewness kurtosis
##   <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>    <dbl>    <dbl>
## 1     2 1501.  20.2    9.3   5.7  16.9  11.2  85.2     15.6     258.
```

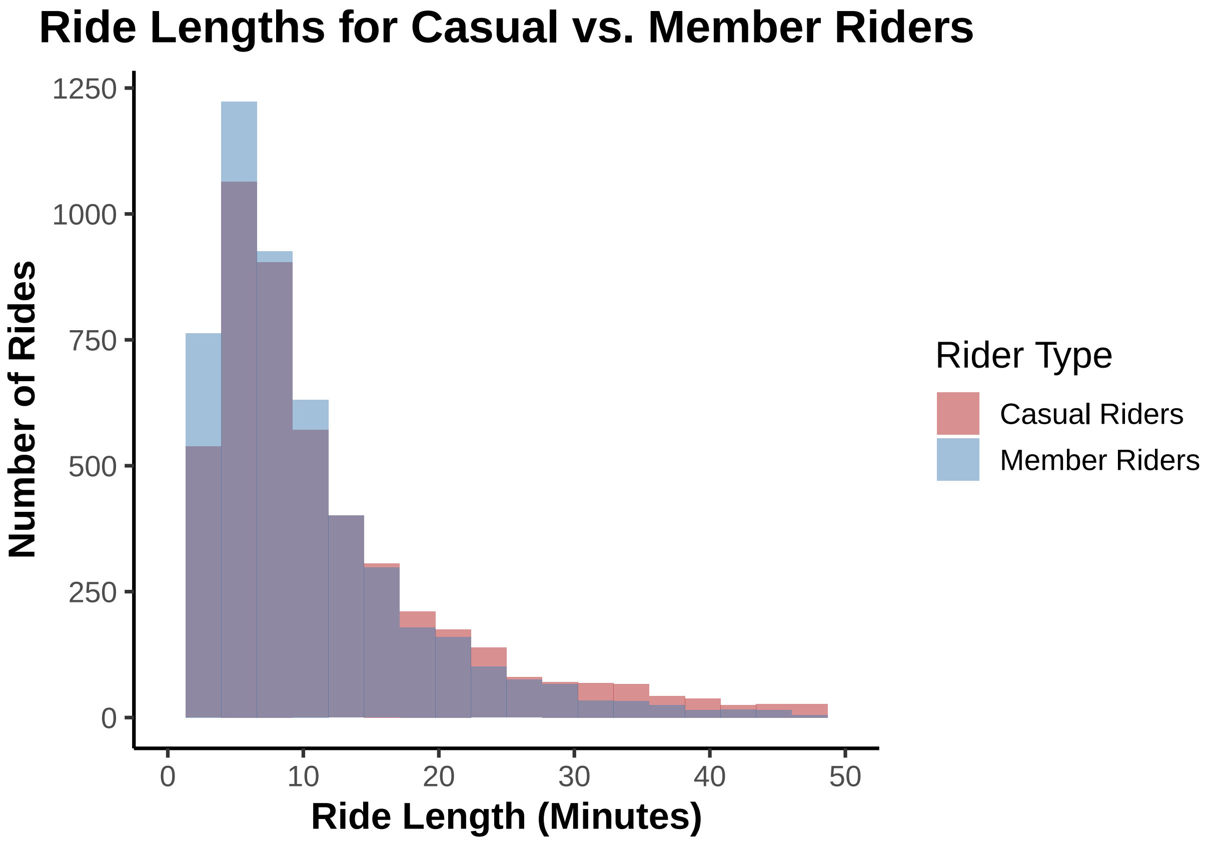
```
# Metrics for member riders
divvy_clean %>%
  filter(member_casual == "member") %>%
  summarise(
    min = min(ride_length),
    max = max(ride_length),
    mean = mean(ride_length),
    median = median(ride_length),
    q1 = quantile(ride_length, 0.25),
    q3 = quantile(ride_length, 0.75),
    iqr = IQR(ride_length),
    std = sd(ride_length),
    skewness = skewness(ride_length),
    kurtosis = kurtosis(ride_length)
  )
```

```
## # A tibble: 1 x 10
##     min   max  mean median    q1    q3   iqr   std skewness kurtosis
##   <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>    <dbl>    <dbl>
## 1     2 1500.  11.5    7.9   4.9  13.1   8.2  30.5     37.2    1662.
```

The following histogram samples the `divvy_clean` data set and takes 5000 samples of both member riders and casual riders. I then visualize them with a histogram in order to see how they compare in terms of ride length. Although the differences are minimal, there are two trends I see from repeated tests, members tend to have slightly higher ride counts for rides under 10 minutes and casual riders tend to have more rides over 20 minutes.

In the plot, the x-axis represents the `ride_length` and the y-axis shows the number of rides. There are two histograms, one for casuals and one for members, both overlapping to see their differences. To do this, I used `fill = member_casual` in conjunction with `position = "identity"` in `geom_histogram()`. Fill made the two histograms and position made them overlap instead of stacking. Last, I used `theme_classic()` and `theme()` to make the plot look cleaner and center / format titles and labels.

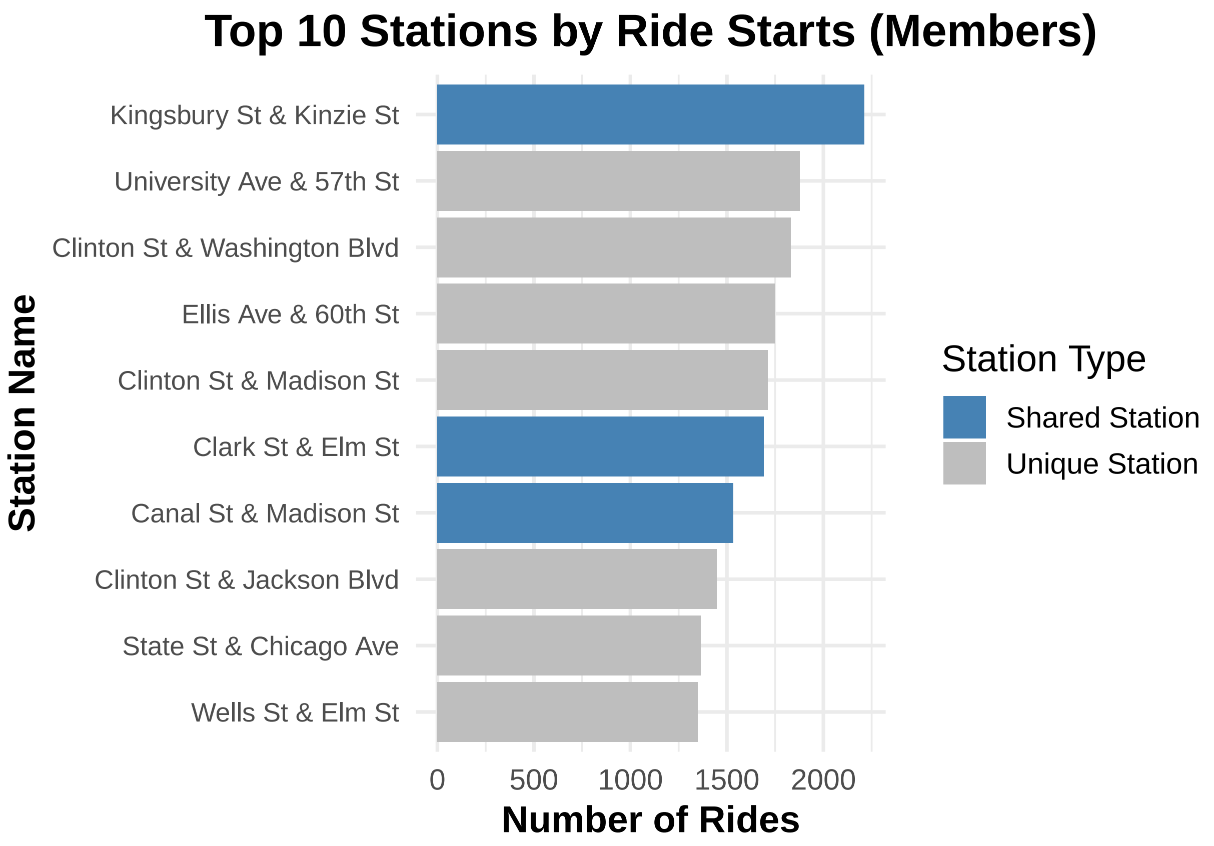
```r
divvy_clean %>%
  group_by(member_casual) %>%
  slice_sample(n = 5000, replace = FALSE) %>%
  ungroup() %>%
  ggplot(aes(x = ride_length, fill = member_casual)) +
  geom_histogram(bins = 20, position = "identity", alpha = 0.5) +
  xlim(c(0, 50)) +
  labs(
    title = "Ride Lengths for Casual vs. Member Riders",
    x = "Ride Length (Minutes)",
    y = "Number of Rides",
    fill = "Rider Type"
  ) +
  scale_fill_manual(
    values = c("casual" = "firebrick", "member" = "steelblue"),
    labels = c("casual" = "Casual Riders", "member" = "Member Riders")
  ) +
  theme_classic(base_size = 14) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title = element_text(face = "bold")
  )
```



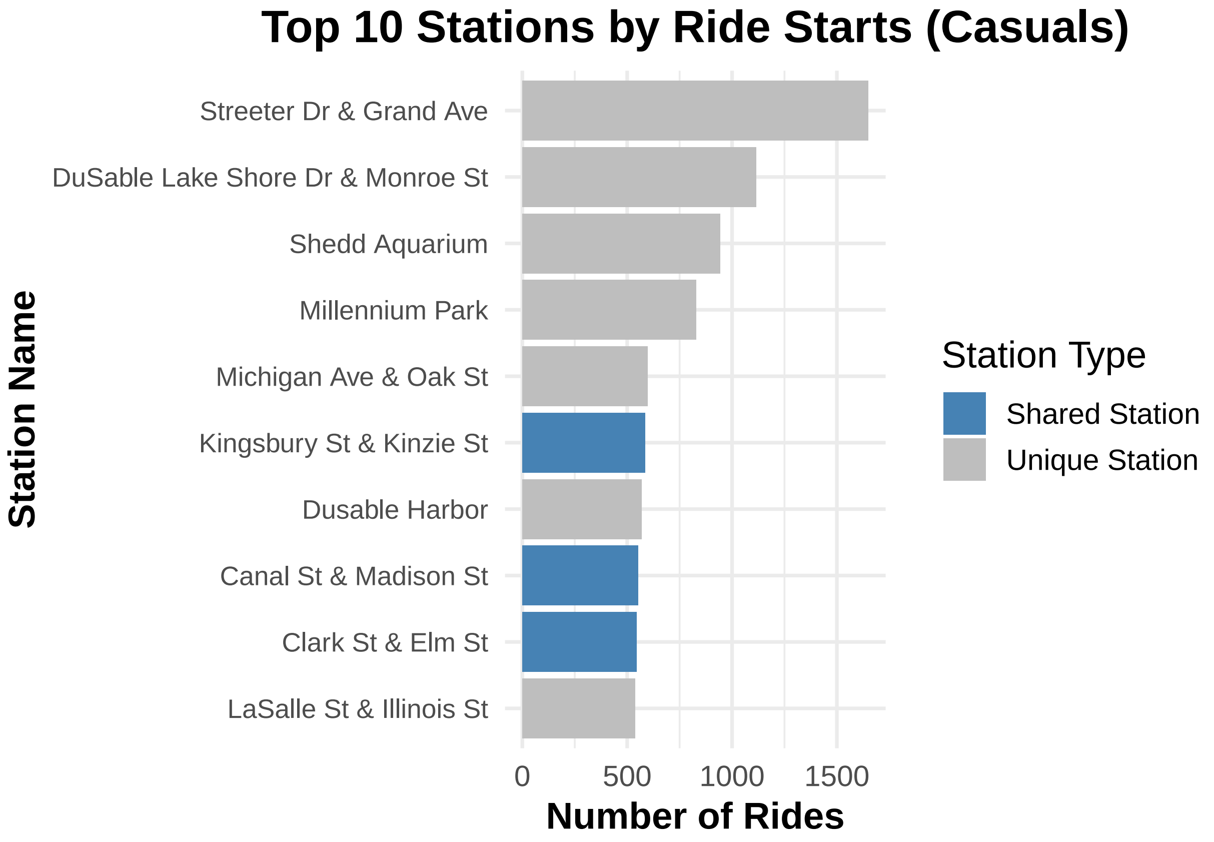
These next two graphs show the top 10 stations with the most rides started at, for members and casual riders. Clearly the members graph has more starts because it takes up roughly 75% of the rides, but there a few differences between the two. First, the members graph, for the most part, gradually decreases as the positions go down, but the casual top 10 have much more drastic differences. In particular, the top 4 out perform the rest by a large margin. This may be due to the fact that the top 4 locations for casual riders seem to be touristy locations such as museums, piers, shopping strips and parks. The top locations for members seem to be at busy locations downtown and near the University of Chicago.

In these plots, the x-axis represents the number of rides while the y-axis represents the top 10 `station_start_name`. In order to find the top 10 for member and casual riders, I first dropped the rows with NA starting stations. Next, I counted the number of each station using `count(start_station_name, sort = TRUE)` and then ordered them in descending order and took the top 10 with `slice_max(n = 10, order_by = n)`. I used `intersect()` to find the common top 10 stations between member and casual riders and created a new variable, shared, that indicates whether a station is shared or unique. As for the plot itself, I used `reorder()` to make sure that the plot uses the top 10 stations by n (starting station count). I also use `coord_flip()`, which flips the x and y axes in make a horizontal bar chart and make the starting stations names more readable. Lastly, I used `scale_fill_manual()`, `labs()`, `theme_minimal()`, and `theme()` in order to add custom colors, titles and labels, make the plot look cleaner, and format the titles and labels.

```r
# Generate data for members and casuals
top_members <- divvy_clean %>%
  drop_na(start_station_name) %>%
  filter(member_casual == "member") %>%
  count(start_station_name, sort = TRUE) %>%
  slice_max(n = 10, order_by = n)

top_casuals <- divvy_clean %>%
  drop_na(start_station_name) %>%
  filter(member_casual == "casual") %>%
  count(start_station_name, sort = TRUE) %>%
  slice_max(n = 10, order_by = n)

# Find shared stations
shared_stations <- intersect(top_members$start_station_name, top_casuals$start_station_name)

# Add a column for shared / unique
top_members <- top_members %>%
  mutate(shared = ifelse(start_station_name %in% shared_stations, "shared", "unique"))

top_casuals <- top_casuals %>%
  mutate(shared = ifelse(start_station_name %in% shared_stations, "shared", "unique"))

# Plot for members
top_members %>%
  ggplot(aes(x = reorder(start_station_name, n), y = n, fill = shared)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  scale_fill_manual(
    values = c("shared" = "steelblue", "unique" = "grey"),
    labels = c("shared" = "Shared Station", "unique" = "Unique Station")
  ) +
  labs(
    title = "Top 10 Stations by Ride Starts (Members)",
    x = "Station Name",
    y = "Number of Rides",
    fill = "Station Type"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title = element_text(face = "bold"),
    axis.text.y = element_text(size = 10)
  )
```



```
# Plot for casuals
top_casuals %>%
  ggplot(aes(x = reorder(start_station_name, n), y = n, fill = shared)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  scale_fill_manual(
    values = c("shared" = "steelblue", "unique" = "grey"),
    labels = c("shared" = "Shared Station", "unique" = "Unique Station")
  ) +
  labs(
    title = "Top 10 Stations by Ride Starts (Casuals)",
    x = "Station Name",
    y = "Number of Rides",
    fill = "Station Type"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title = element_text(face = "bold"),
    axis.text.y = element_text(size = 10)
  )
```

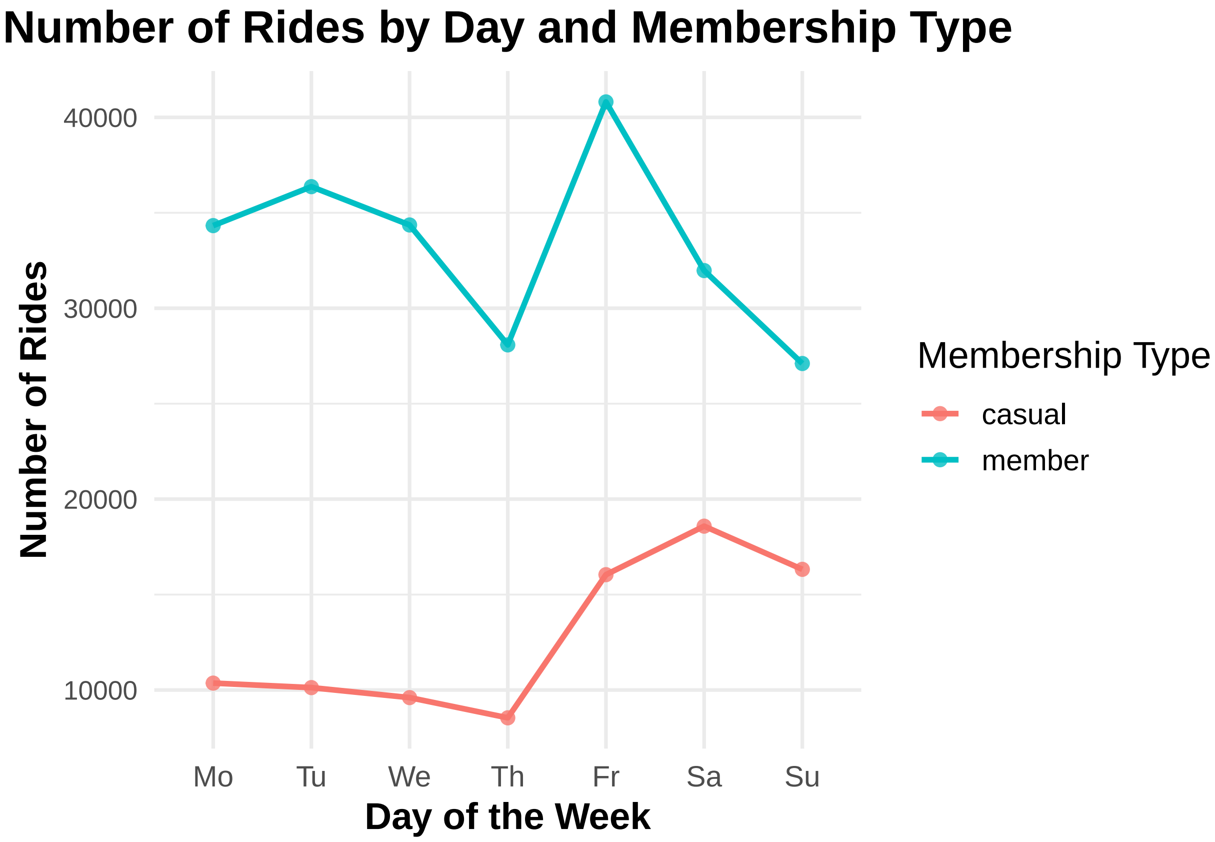


The last visual is a line plot that shows the number of rides taken per each day of the week for both member and casual riders. Looking at the red line (casual riders), Friday, Saturday and Sunday have a much higher ride count, ranging from about 50% to over 80% more rides than the rest of the week days. As for the blue line (member riders), the data shows Friday is the most popular day for rides while Thursday and Sunday have the lowest counts. In both groups, Thursdays are one of the lowest.

In this plot, the x-axis represents the `day_of_week` while the y-axis represents the number of rides per each day. My first step in creating this visualization was to group the data by day of the week using `group_by(day_of_week, member_casual)`. Then, I used `summarize()` and created a count variable for each day of the week. Next, I used `mutate()` and `factor()` to change the order of the week to be Monday to Sunday. Finally, I used `labs()`, `theme_minimal()` and `theme()` to add a title and labels, create a cleaner look, and adjust the position / formatting of the title / labels.

```
divvy_clean %>%
  group_by(day_of_week, member_casual) %>%
  summarize(count = n(), .groups = "drop") %>%
  mutate(day_of_week = factor(day_of_week, levels = c("Monday", "Tuesday", "Wednesday", "Thursday", "Fr
  ggplot(aes(x = day_of_week, y = count, color = member_casual, group = member_casual)) +
  geom_line(size = 1) +
  geom_point(size = 2, alpha = 0.8) +
  labs(
    title = "Number of Rides by Day and Membership Type",
    x = "Day of the Week",
    y = "Number of Rides",
    color = "Membership Type"
  ) +
  scale_x_discrete(labels = c("Mo", "Tu", "We", "Th", "Fr", "Sa", "Su")) +
  theme_minimal(base_size = 14) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title = element_text(face = "bold"),
    axis.text.y = element_text(size = 10)
  )
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



Step 4: Analyze

The analyze step of data analysis involves taking the findings from the process section (visualizations, trends, patterns, statistics) and interpreting them. This stage also utilizes comparisons, looking at outliers / anomalies, and the implications of these findings.

Interpreting the Findings

Distribution of Casual vs. Member Riders

The first bar chart shows that just over a quarter of the recorded rides were casual riders using a day pass or one-time ride. The remaining, nearly three-quarters, are rides from members. There is a clear difference in usage between casual and member rides, likely due to the fact that member riders have a higher chance of being commuters who frequently use the bikes, while casual riders may lean more towards tourists or leisurely activity.

Ride Type Differences

In the second bar chart, we see the distribution of ride type for both casual and member riders. Both groups have similar results with classic bikes being the bike of choice ~45% of the time and electric bikes being the choice ~55% of the time. This shows the no matter the rider, they tend to choose the same bike types roughly at the same proportion.

Ride Durations

Member riders tend to have more slightly more rides than casual riders in the sub-10 minute category. On the other hand, casual riders tend to have sizably more rides in the 20+ minute category as compared to member riders. This can reasonably be explained by members taking smaller rides for commutes / small distance rides since they pay annually, while casual riders are more likely to use it in leisurely activities or longer bike rides since they only pay for a single ride / day pass.

Top Stations

The difference in the top starting stations highlights the difference in usage between casual and member riders. The top stations for casual riders are located at docks, piers, museums, and parks. Again, this is likely due to casual riders being tourists or for recreational activities. The top stations for member riders are located throughout downtown Chicago and near the University of Chicago. This points towards many members riding for commutes or similar.

Busiest Days of the Week

From the final line plot, it is clear that the casual riders (red line) tend to ride at a much higher rate on Fridays and the weekends compared to the remaining week days. For member riders, the day with the highest number of rides are Fridays. In both groups, Thursday tends to have one of, if not the lowest, number of rides throughout the week.

Drawing Conclusions

To sum up the findings, roughly 25% of rides are from casual riders while the remaining 75% are from member riders. The casual riders tend to ride more on Friday, Saturday, and Sunday, while members don't have a clear pattern, only that Fridays are the most popular day to ride. Member riders tend to have more rides than casual riders under 10 mins, but for rides 20 minutes or more, casual riders have a much larger count than member riders. This is likely due the fact that member riders are more probable to be commuters or use the bikes for short rides since they don't pay for each ride individually. For the casual riders, it is more probable for them to use the bikes for longer durations, wanting to get their money's worth since they pay for just a single ride or day pass. To further push this point, the members most popular starting locations were around downtown and the University of Chicago while the casual riders top 4 most popular were all leisurely / prominent tourist locations.

Step 5: Share

This stage of the data analysis involves taking my findings from the process and analyze stages and putting together a slideshow or presentation for the higher ups / stakeholders involved in the business task. The most important thing to keep in mind when creating your presentation is who your audience is. It is important be aware of how high / low level you need to be. As the situation for this case study has no one to present to, there is no work to be done for this step, however, we were given our business task from the 'director of marketing' at Cyclistic. With this information, I can assume (unless told otherwise by the director) a high-level overview will suffice for the presentation, focusing own their director's wants.

Step 6: Act

This is the final step of the data analysis process. At this point, you want to take your findings and translate them into actionable recommendations for your audience. While my assigned business task is 'How do annual members and casual riders use Cyclistic bikes differently', I know that the analysis as a whole aims to **design marketing strategies aimed at converting casual riders into annual members**. My findings on this data don't necessarily translate into an viable path for this goal, but I can give some possible actions:

- Set up advertisement, listing the most appealing benefits of buying the annual membership, at the most popular starting and ending stations for casual riders.
- Look at a partnership with the nearby businesses of the most popular stations for casual riders and create some kind of discount / perks that only members can obtain.
- While the data doesn't contain this information, assuming that a respectable number of the casual riders are residents of Chicago, create free trial, possibly for a month, or a discounted first year to gain new members.