Cyclistic Bike-Sharing Data Analysis

Thanakorn Thanakraikiti (Nun)

June 21, 2023

Contents

1 Intoduction

Cyclistic is a successful American bicycle-sharing program that was established in 2016. Since then, the program has grown to a fleet of 5,824 bicycles that are geotracked and locked into a network of 692 stations across Chicago. The bikes can be unlocked from one station and returned to any other station in the system at any time. Cyclistic users are more likely to ride for leisure, but approximately 30% use them to commute to work each day.

Cyclistic's marketing strategy has primarily focused on building general awareness and appealing to broad consumer segments. The program offers a variety of pricing plans, including single-ride passes, full-day passes, and annual memberships. Cyclistic classifies its riders into the following groups based on these plans: *casual riders* (users who purchase single-ride passes or full-day passes) and *annual members* (users who purchase an annual membership).

Cyclistic's flexible pricing plans attract a larger customer base, but financial analysts have determined that annual members are more profitable. However, casual riders are already aware of the Cyclistic program and have chosen Cyclistic to meet their mobility needs. This suggests that a marketing campaign that targets existing customers is likely to be more effective at expanding the business than a campaign that targets only new customers.

Therefore, Cyclistic's marketing analytics team is interested in understanding how casual riders and annual members use Cyclistic bikes differently. By understanding these differences, the marketing analytics team can develop more targeted marketing strategies to convert casual riders into annual members.

1.1 Purpose

This report analyzes Cyclistic's historical bike trip data from 2022 in order to identify how annual members and casual riders use Cyclistic bikes differently. The report aims to extract insights and develop the most appropriate marketing strategies that appeal to casual riders and encourage them to subscribe to annual memberships. This will enable the business to improve productivity and respond to economic changes in a productive and efficient manner. The report outlines the behavioral differences in Cyclistic bike usage between annual members and casual riders. The report further demonstrates that the recommended solutions will help Cyclistic to expand its business growth.

1.2 Stakeholders

Key stakeholders include:

- Marketing analytics team who collect, analyze, and report data to help guide Cyclistic's marketing strategy
- Marketing manager who supports the marketing campaign to promote Cyclistic's bike-share program
- Executive team who approve the recommended marketing program

2 Data Sources

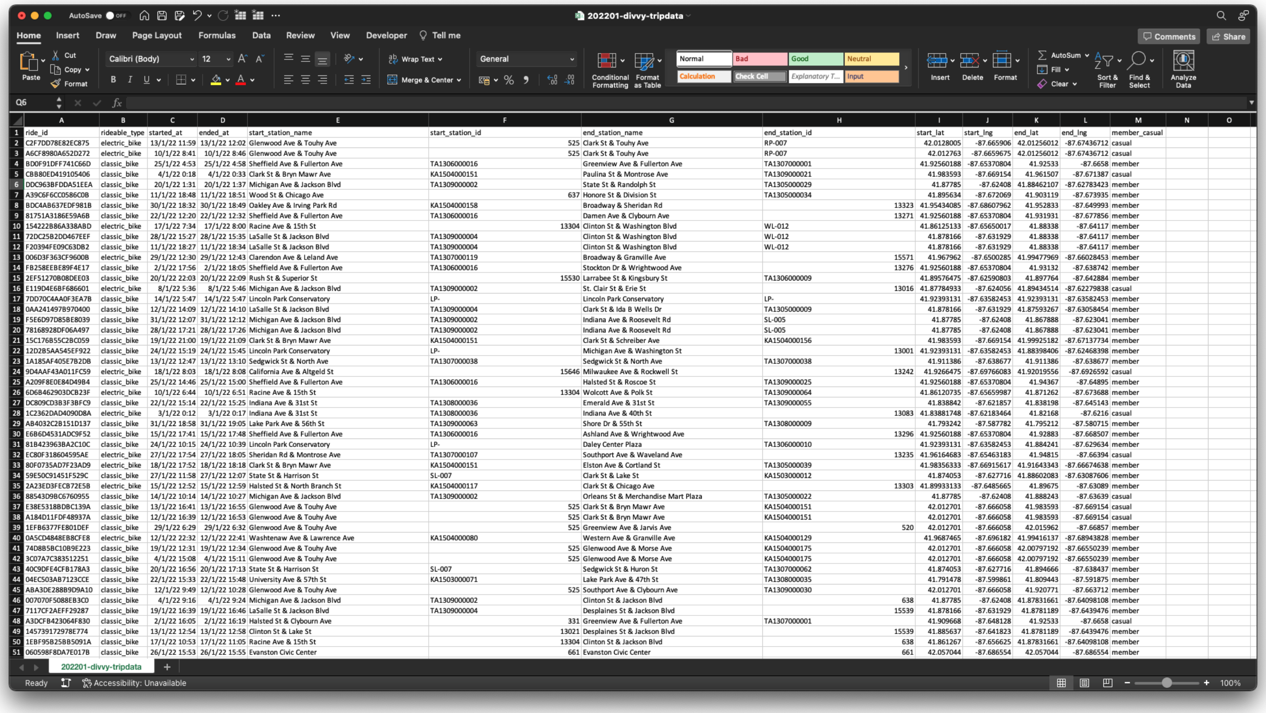
This report uses Cyclistic's historical trip data from January 2022 to December 2022 as its primary source. The data sets can be downloaded from the company's website, which can be find in the following details:

- Data Source: Trip History Data
- Data Website: System Data Website
- License: Data License Agreement

The data sets are collected by the City of Chicago (a government agency) and published by Cyclistic. They contain historical trip data for all of the company's customer (rider) data since 2013, including the following information: ride history, the location of stations for each ride, and other relevant information

2.1 Data Description

The data set is stored in twelve CSV files, one for each month of 2022 (*202201-divvy-tripdata.csv* through *202212-divvy-tripdata.csv*). Each file contains a set of fields (columns) and records (rows). Each record represents a single bike trip, which is uniquely identified by an identifier for the ride.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	ride_id	rideable_type	started_at	ended_at	start_station_name	start_station_id	end_station_name	end_station_id	start_lat	start_lng	end_lat	end_lng	member_casual
2	C2F7DD78E82EC875	electric_bike	13/1/22 11:59	13/1/22 12:02	Glenwood Ave & Touhy Ave	525	Clark St & Touhy Ave	RP-007	42.0128005	-87.665906	42.01256012	-87.67436712	casual
3	A6CF8980A652D272	electric_bike	10/1/22 8:41	10/1/22 8:46	Glenwood Ave & Touhy Ave	525	Clark St & Touhy Ave	RP-007	42.012763	-87.6659675	42.01256012	-87.67436712	casual
4	BD0F91DFF741C66D	classic_bike	25/1/22 4:53	25/1/22 4:58	Sheffield Ave & Fullerton Ave	TA1306000016	Greenview Ave & Fullerton Ave	TA1307000001	41.92560188	-87.65370804	41.92533	-87.6658	member
5	CBB80ED419105406	classic_bike	4/1/22 0:18	4/1/22 0:33	Clark St & Bryn Mawr Ave	KA1504000151	Paulina St & Montrose Ave	TA1309000021	41.983593	-87.669154	41.961507	-87.671387	casual
6	DDC963BFDDA51EEA	classic_bike	20/1/22 1:31	20/1/22 1:37	Michigan Ave & Jackson Blvd	TA1309000002	State St & Randolph St	TA1305000029	41.87785	-87.62408	41.88462107	-87.62783423	member
7	A39C6F6CC0586C0B	classic_bike	11/1/22 18:48	11/1/22 18:51	Wood St & Chicago Ave	637	Honore St & Division St	TA1305000034	41.895634	-87.672069	41.903119	-87.673935	member
8	BDC4AB637EDF981B	classic_bike	30/1/22 18:32	30/1/22 18:49	Oakley Ave & Irving Park Rd	KA1504000158	Broadway & Sheridan Rd	13323	41.95434085	-87.68607962	41.952833	-87.649993	member
9	81751A3186E59A6B	classic_bike	22/1/22 12:20	22/1/22 12:32	Sheffield Ave & Fullerton Ave	TA1306000016	Damen Ave & Clybourn Ave	13271	41.92560188	-87.65370804	41.931931	-87.677856	member
10	154222B86A338ABD	electric_bike	17/1/22 7:34	17/1/22 8:00	Racine Ave & 15th St	13304	Clinton St & Washington Blvd	WL-012	41.86125133	-87.65650017	41.88338	-87.64117	member
11	72DC25B2DD467EEF	classic_bike	28/1/22 15:27	28/1/22 15:35	LaSalle St & Jackson Blvd	TA1309000004	Clinton St & Washington Blvd	WL-012	41.878166	-87.631929	41.88338	-87.64117	member
12	F20394FE09C63DB2	classic_bike	11/1/22 18:27	11/1/22 18:34	LaSalle St & Jackson Blvd	TA1309000004	Clinton St & Washington Blvd	WL-012	41.878166	-87.631929	41.88338	-87.64117	member
13	006D3F363CF9600B	electric_bike	29/1/22 12:30	29/1/22 12:43	Clarendon Ave & Leland Ave	TA1307000119	Broadway & Granville Ave	15571	41.967962	-87.6500285	41.99477969	-87.66028453	member
14	FB258EEBE89F4E17	classic_bike	2/1/22 17:56	2/1/22 18:05	Sheffield Ave & Fullerton Ave	TA1306000016	Stockton Dr & Wrightwood Ave	13276	41.92560188	-87.65370804	41.93132	-87.638742	member
15	2EF51270B08DEE03	classic_bike	20/1/22 22:03	20/1/22 22:09	Rush St & Superior St	15530	Larrabee St & Kingsbury St	TA1306000009	41.89576475	-87.62590803	41.897764	-87.642884	member
16	E119D4E6BF686601	electric_bike	8/1/22 5:36	8/1/22 5:46	Michigan Ave & Jackson Blvd	TA1309000002	St. Clair St & Erie St	13016	41.87784933	-87.624056	41.89434514	-87.62279838	casual
17	7DD70C4AA0F3EA7B	classic_bike	14/1/22 5:47	14/1/22 5:47	Lincoln Park Conservatory	LP-	Lincoln Park Conservatory	LP-	41.92393131	-87.63582453	41.92393131	-87.63582453	member
18	0AA241497B970400	classic_bike	12/1/22 14:09	12/1/22 14:10	LaSalle St & Jackson Blvd	TA1309000004	Clark St & Ida B Wells Dr	TA1305000009	41.878166	-87.631929	41.87593267	-87.63058454	member
19	F5E6D97D85BE8039	classic_bike	31/1/22 12:07	31/1/22 12:12	Michigan Ave & Jackson Blvd	TA1309000002	Indiana Ave & Roosevelt Rd	SL-005	41.87785	-87.62408	41.867888	-87.623041	member
20	78168928DF06A497	classic_bike	28/1/22 17:21	28/1/22 17:26	Michigan Ave & Jackson Blvd	TA1309000002	Indiana Ave & Roosevelt Rd	SL-005	41.87785	-87.62408	41.867888	-87.623041	member
21	15C176B55C2BC059	classic_bike	19/1/22 21:00	19/1/22 21:09	Clark St & Bryn Mawr Ave	KA1504000151	Clark St & Schreiber Ave	KA1504000156	41.983593	-87.669154	41.99925182	-87.67137734	member
22	12D2B5AA545EF922	classic_bike	24/1/22 15:19	24/1/22 15:45	Lincoln Park Conservatory	LP-	Michigan Ave & Washington St	13001	41.92393131	-87.63582453	41.88398406	-87.62468398	member
23	1A185AF405E7B2DB	classic_bike	13/1/22 12:47	13/1/22 13:10	Sedgwick St & North Ave	TA1307000038	Sedgwick St & North Ave	TA1307000038	41.911386	-87.638677	41.911386	-87.638677	member
24	9D4AAF43A011FC59	electric_bike	18/1/22 8:03	18/1/22 8:08	California Ave & Altgeld St	15646	Milwaukee Ave & Rockwell St	13242	41.9266475	-87.69766083	41.92019556	-87.6926592	casual
25	A209F8E0E84D49B4	classic_bike	25/1/22 14:46	25/1/22 15:00	Sheffield Ave & Fullerton Ave	TA1306000016	Halsted St & Roscoe St	TA1309000025	41.92560188	-87.65370804	41.94367	-87.64895	member
26	6D6B462903DCB23F	electric_bike	10/1/22 6:44	10/1/22 6:51	Racine Ave & 15th St	13304	Wolcott Ave & Polk St	TA1309000064	41.86120735	-87.65659987	41.871262	-87.673688	member
27	DC809CD3B3F3BFC9	classic_bike	22/1/22 15:14	22/1/22 15:25	Indiana Ave & 31st St	TA1308000036	Emerald Ave & 31st St	TA1309000055	41.838842	-87.621857	41.838198	-87.645143	member
28	1C2362DAD4090D8A	electric_bike	3/1/22 0:12	3/1/22 0:17	Indiana Ave & 31st St	TA1308000036	Indiana Ave & 40th St	13083	41.83881748	-87.62183464	41.82168	-87.6216	casual
29	AB4032C2B151D137	classic_bike	31/1/22 18:58	31/1/22 19:05	Lake Park Ave & 56th St	TA1309000063	Shore Dr & 55th St	TA1308000009	41.793242	-87.587782	41.795212	-87.580715	member
30	E6B6D4531ADC9F52	classic_bike	15/1/22 17:41	15/1/22 17:48	Sheffield Ave & Fullerton Ave	TA1306000016	Ashland Ave & Wrightwood Ave	13296	41.92560188	-87.65370804	41.92883	-87.668507	member
31	81B423963BA2C10C	classic_bike	24/1/22 10:15	24/1/22 10:39	Lincoln Park Conservatory	LP-	Daley Center Plaza	TA1306000010	41.92393131	-87.63582453	41.884241	-87.629634	member
32	EC80F318604595AE	electric_bike	27/1/22 17:54	27/1/22 18:05	Sheridan Rd & Montrose Ave	TA1307000107	Southport Ave & Waveland Ave	13235	41.96164683	-87.65463183	41.94815	-87.66394	casual
33	80F0735AD7F23AD9	electric_bike	18/1/22 17:52	18/1/22 18:18	Clark St & Bryn Mawr Ave	KA1504000151	Elston Ave & Cortland St	TA1305000039	41.98356333	-87.66915617	41.91643343	-87.66674638	member
34	59E50C91451F529C	classic_bike	27/1/22 11:58	27/1/22 12:07	State St & Harrison St	SL-007	Clark St & Lake St	KA1503000012	41.874053	-87.627716	41.88602083	-87.63087606	member
35	2A23ED3FECB72E5B	electric_bike	15/1/22 12:52	15/1/22 12:59	Halsted St & North Branch St	KA1504000117	Clark St & Chicago Ave	13303	41.89933133	-87.6485665	41.89675	-87.63089	member
36	88543D9BC6760955	classic_bike	14/1/22 10:14	14/1/22 10:27	Michigan Ave & Jackson Blvd	TA1309000002	Orleans St & Merchandise Mart Plaza	TA1305000022	41.87785	-87.62408	41.888243	-87.63639	casual
37	E38E5318BDBC139A	classic_bike	13/1/22 16:41	13/1/22 16:55	Glenwood Ave & Touhy Ave	525	Clark St & Bryn Mawr Ave	KA1504000151	42.012701	-87.666058	41.983593	-87.669154	casual
38	A184D11FDF48937A	classic_bike	12/1/22 16:39	12/1/22 16:53	Glenwood Ave & Touhy Ave	525	Clark St & Bryn Mawr Ave	KA1504000151	42.012701	-87.666058	41.983593	-87.669154	casual
39	1EFB6377FE801DEF	classic_bike	29/1/22 6:29	29/1/22 6:32	Glenwood Ave & Touhy Ave	525	Greenview Ave & Jarvis Ave	520	42.012701	-87.666058	42.015962	-87.66857	member
40	0A5CD4848EB8CFE8	electric_bike	12/1/22 22:32	12/1/22 22:41	Washtenaw Ave & Lawrence Ave	KA1504000080	Western Ave & Granville Ave	KA1504000129	41.9687465	-87.696182	41.99416137	-87.68943828	member
41	74D8B5BC10B9E223	classic_bike	19/1/22 12:31	19/1/22 12:34	Glenwood Ave & Touhy Ave	525	Glenwood Ave & Morse Ave	KA1504000175	42.012701	-87.666058	42.00797192	-87.66550239	member
42	3C07A7C383512251	classic_bike	4/1/22 15:08	4/1/22 15:11	Glenwood Ave & Touhy Ave	525	Glenwood Ave & Morse Ave	KA1504000175	42.012701	-87.666058	42.00797192	-87.66550239	member
43	40C9DFE4CFB178A3	classic_bike	20/1/22 16:56	20/1/22 17:13	State St & Harrison St	SL-007	Sedgwick St & Huron St	TA1307000062	41.874053	-87.627716	41.894666	-87.638437	member
44	04EC503AB7123CCE	classic_bike	22/1/22 15:33	22/1/22 15:48	University Ave & 57th St	KA1503000071	Lake Park Ave & 47th St	TA1308000035	41.791478	-87.599861	41.809443	-87.591875	member
45	ABA3DE288B9D9A10	classic_bike	12/1/22 9:49	12/1/22 10:28	Glenwood Ave & Touhy Ave	525	Southport Ave & Clybourn Ave	TA1309000030	42.012701	-87.666058	41.920771	-87.663712	member
46	007070F5088EB3C0	classic_bike	4/1/22 9:16	4/1/22 9:24	Michigan Ave & Jackson Blvd	TA1309000002	Clinton St & Jackson Blvd	638	41.87785	-87.62408	41.87831661	-87.64098108	member
47	7117CF2AEFF29287	classic_bike	19/1/22 16:39	19/1/22 16:46	LaSalle St & Jackson Blvd	TA1309000004	Desplaines St & Jackson Blvd	15539	41.878166	-87.631929	41.8781189	-87.6439476	member
48	A3DCFB423064F830	classic_bike	2/1/22 16:05	2/1/22 16:19	Halsted St & Clybourn Ave	331	Greenview Ave & Fullerton Ave	TA1307000001	41.909668	-87.648128	41.92533	-87.6658	casual
49	145739172978E774	classic_bike	13/1/22 12:54	13/1/22 12:58	Clinton St & Lake St	13021	Desplaines St & Jackson Blvd	15539	41.885637	-87.641823	41.8781189	-87.6439476	member
50	1EBF95B25BB5091A	classic_bike	17/1/22 10:53	17/1/22 11:05	Racine Ave & 15th St	13304	Clinton St & Jackson Blvd	638	41.861267	-87.656625	41.87831661	-87.64098108	member
51	060598F8DA7E017B	classic_bike	26/1/22 15:53	26/1/22 15:55	Evanston Civic Center	661	Evanston Civic Center	661	42.057044	-87.686554	42.057044	-87.686554	member

Figure 1: The spreadsheet example, containing records of Cyclistic's historical trip data in January 2022

The data sets contain the following fields:

- `ride_id` : A unique identifier for the ride
- `rideable_type` : The type of bike for the ride (classic bike, electric bike or docked bike)
- `started_at` : The date and time at which the ride started
- `ended_at` : The date and time at which the ride ended

- `start_station_id` : The ID of the station where the ride started
- `start_station_name` : The name of the station where the ride started
- `end_station_id` : The ID of the station where the ride ended
- `end_station_name` : The name of the station where the ride ended
- `start_lat` : The latitude of the station where the ride started
- `start_lng` : The longitude of the station where the ride started
- `end_lat` : The latitude of the station where the ride ended
- `end_lng` : The longitude of the station where the ride ended
- `member_casual` : The type of user who took the ride (member or casual)

2.2 Data Credibility

The data is evaluated for bias and credibility using the ROCCC method:

- **Reliable**: The data is considered reliable because it is open data provided by Cyclistic and the City of Chicago. It is a comprehensive and accurate dataset of Cyclistic's historical bike trips. The data includes information on the start and end location of each trip, the time of day the trip was taken, the type of user (casual or annual member), and the length of the trip. There are some missing values in some fields, but these can be used to explore how different customer types use Cyclistic bikes.
- **Original**: The data is considered to be original, based on a primary source of information from the company.
- **Comprehensive**: It is important to note that Cyclistic is subject to data privacy restrictions that prohibit the use of riders' personally identifiable information (PII). This means that the company cannot connect pass purchases to credit card numbers to determine if casual riders live in the Cyclistic service area or if they have purchased multiple single passes, as well as identify the rider's gender and age. Despite these restrictions, the data is complete and contains all the information needed to understand how members and casual riders use Cyclistic bikes. This includes information such as the number of rides taken, the length of each ride, the start and end stations of each ride, the time of day each ride was taken.
- **Current**: The data is current and updated monthly. The selected datasets are the most recent data available.
- **Cited**: The data is properly cited as it contains the dataset name, author, and published year.

Overall, the Cyclistic bike data is considered a credible source for analysis based on the Reliability, Originality, Comprehensiveness, Currency, and Citation (ROCCC) method. The data can be used to explore how different customer types use Cyclistic bikes. For example, the data can be used to determine which customer segments are most likely to use Cyclistic bikes for commuting, recreation, or transportation. The data can also be used to identify trends in bike usage, such as the time of day when bikes are most likely to be used or the types of bikes that are most popular.

While there are some restrictions on rider's personal information, there is still a great deal of information that can be learned from it. However, the data needs to be processed before analysis can be performed. This includes cleaning the data to remove errors and inconsistencies, and transforming the data into a format that is suitable for analysis. The data processing will be performed in the next section.

3 Data Cleaning and Manipulation

This section documents the data cleaning and manipulation. It includes the following tasks:

1. Setting up and loading data into R environment
2. Removing duplicate or irrelevant observations
3. Fixing structural errors
4. Handle missing data
5. Adding new columns
6. Filter unwanted outliers

This report uses the ***R programming language*** in conjunction with the ***RStudio*** integrated development environment (IDE) to complete these tasks. All R scripts are available on the GitHub repository.

3.1 Setting Up and Loading Data into R Environment

Prior to cleaning the data, it is important to set up the R environment on the project. This requires loading the following packages into the environment:

- `tidyverse` for importing and wrangling the data
- `janitor` for cleaning data
- `ggplot2` for visualizing the data
- `ggmap` for visualizing the data in a map

Once the environment was set up, each CSV file was imported into a separate data frame, and then these data frames were combined into a single data frame, named *trip_data_v2*. Upon initial inspection, the data frame contains 5,667,717 rows and 13 columns. The table below provides a preview of the *trip_data_v2* data frame.

Table 1: A preview of Cyclistic's historical trip data

ride_id	98D355D9A9852BE9
rideable_type	classic_bike
started_at	2022-01-01 00:00:05
ended_at	2022-01-01 00:01:48
start_station_name	Michigan Ave & 8th St
start_station_id	623
end_station_name	Michigan Ave & 8th St
end_station_id	623
start_lat	41.87277
start_lng	-87.62398
end_lat	41.87277
end_lng	-87.62398
member_casual	casual

3.2 Removing Duplicate or Irrelevant Observations

The first task in data cleaning and manipulation is to remove unwanted observations from the data set, including duplicate observations or irrelevant observations. Duplicate observations will happen most often during the data collection. This task involves two main parts, including removing irrelevant columns and remove duplicate values.

Initially, irrelevant columns were removed from the data set as they were not required for this analysis. This was done by using the `select()` function to remove the following columns: `start_station_id` and `end_station_id`. After that, it is important to remove duplicate records. To detect duplicate records in the `ride_id` column, the `summarize()` function was used to count the number of rows and unique rows in the data set. This function returned a data frame with two columns: `total_ride` (the number of rows) and `unique_ride` (the number of unique rows).

Table 2: The total number of rides and unique rides in Cyclistic's historical trip data

total_ride	unique_ride
5667717	5667717

The result shows that there are 5,667,717 rides in total and 5,667,717 unique rides in the data set. This implies that the data set does not contain any duplicate values. Therefore, the task of removing duplicate records can be skipped in the `ride_id` column. There are currently 5,667,717 rows and 11 columns remaining in the data set.

3.3 Fixing Structural Errors

The next task is to fix structural errors occurred in the data. This involves ensuring that all data is in a consistent formats or naming conventions, with the same data types and values. This includes bike and rider types and station names.

3.3.1 Bike and Rider Types

This step involved checking for irrelevant records in the `rideable_type` and `member_casual` columns. The `count()` function was used to count the number of records in each category.

Table 3: A list of bike type in in Cyclistic's historical trip data

rideable_type	total_ride
classic_bike	2601214
docked_bike	177474
electric_bike	2889029

Table 4: A list of user type in in Cyclistic's historical trip data

member_casual	total_ride
casual	2322032
member	3345685

Tables 3 and 4 show that there are three types of bikes: classic_bike, docked_bike, and electric_bike. There are also two types of riders: casual and member. These results suggest that there are no consistency errors in the `rideable_type` or `member_casual` columns.

3.3.2 Station Names

This step involved making the station names consistent in the `start_station_name` and `end_station_name` columns. The `count()` function was used again to ensure that there were no inconsistencies. The pre-cleaned versions of the `start_station_name` and `end_station_name` columns can be found in the *preclean-start-station.csv* and *preclean-end-station.csv*.

However, an inspection of these columns revealed some inconsistent formats. First, the data set included test stations, which should not be included in the analysis and must be removed. There were eight test stations, including `Pawel Bialowas - Test- PBSC charging station`, `Hastings WH 2`, `DIVVY CASSETTE REPAIR MOBILE STATION`, `Base - 2132 W Hubbard Warehouse`, `Base - 2132 W Hubbard`, `NewHastings`, `WestChi`, and `WEST CHI-WATSON`. The test stations were removed using the following R code:

```
test_station_list <- c("Pawel Bialowas - Test- PBSC charging station",
                       "Hastings WH 2",
                       "DIVVY CASSETTE REPAIR MOBILE STATION",
                       "Base - 2132 W Hubbard Warehouse",
                       "Base - 2132 W Hubbard",
                       "NewHastings",
                       "WestChi",
                       "WEST CHI-WATSON")

trip_data_v2 <- trip_data_v2[
  !( trip_data_v2$start_station_name %in% test_station_list |
     trip_data_v2$end_station_name %in% test_station_list ), ]
```

Additionally, it was found that the station names contained some extraneous characters and words. These extraneous characters and words could cause problems when grouping the data by station name, so they must be removed. The extraneous characters and words that were found include `*`, `- Charging`, `(Temp)`, and `amp;`. The following R code was used to remove these extraneous characters and words:

```r
trip_data_v2 <- trip_data_v2 %>%
  mutate(start_station_name = str_replace_all(start_station_name, fixed("*"), "")) %>%
  mutate(start_station_name = str_replace_all(start_station_name, fixed(" - Charging"), "")) %>%
  mutate(start_station_name = str_replace_all(start_station_name, fixed(" (Temp)"), "")) %>%
  mutate(start_station_name = str_replace_all(start_station_name, fixed("amp;"), "")) %>%
  mutate(end_station_name = str_replace_all(end_station_name, fixed("*"), "")) %>%
  mutate(end_station_name = str_replace_all(end_station_name, fixed(" - Charging"), "")) %>%
  mutate(end_station_name = str_replace_all(end_station_name, fixed(" (Temp)"), "")) %>%
  mutate(end_station_name = str_replace_all(end_station_name, fixed("amp;"), ""))
```

Finally, the task of fixing structural errors has been completed. The data set now contains 5,665,349 rows and 11 columns.

3.4 Handle Missing Data

The third task in data cleaning and manipulation is to handle missing data. There are a few approaches to dealing with missing data, which include:

1. **Imputation:** This involves filling in the missing values with estimates based on other observations.
2. **Deletion:** This involves dropping the observations that have missing values.

Neither of these approaches is ideal, but both can be considered. The following table shows a list of the total number of missing values in each column.

Table 5: A list of the total number of missing values in each column (before removing missing values)

	total_na
ride_id	0
rideable_type	0
started_at	0
ended_at	0
start_station_name	833025
end_station_name	891896
start_lat	0
start_lng	0
end_lat	5858

	total_na
end_lng	5858
member_casual	0

The table above shows that there are four columns with missing values, including:

- `start_station_name` (833,025 rows)
- `end_station_name` (891,896 rows)
- `end_lat` (5,858 rows)
- `end_lng` (5,858 rows)

In the next step, the missing values in the `start_station_name` and `end_station_name` columns will be replaced with the other observations based on geographic coordinates. The remaining missing values will be removed. The following R code was used to impute the missing station names with geographic coordinates:

```r
trip_data_v2 <- trip_data_v2 %>%
    group_by(start_lat, start_lng) %>%
    fill(start_station_name, .direction = "downup") %>%
    ungroup()

trip_data_v2 <- trip_data_v2 %>%
    group_by(end_lat, end_lng) %>%
    fill(end_station_name, .direction = "downup") %>%
    ungroup()
```

The number of rows with missing values in the `start_station_name` and `end_station_name` columns is reduced to 476,854 and 559,273, respectively. The remaining missing values (including `end_lat` and `end_lng`) are then removed by running the following R code:

```r
trip_data_v2 <- trip_data_v2[complete.cases(trip_data_v2), ]
```

Missing data was completely removed from the data frame. After inspecting the station names again, all missing and inconsistent values were eliminated. The post-cleaned versions of the `start_station_name` and `end_station_name` columns can also be found in the *postclean-start-station.csv* and *postclean-end-station.csv* files. As shown in the table below, there are currently no missing values in the data frame.

Table 6: A list of the total number of missing values in each column (after removing missing values)

	total_na
ride_id	0
rideable_type	0
started_at	0
ended_at	0
start_station_name	0
end_station_name	0
start_lat	0
start_lng	0
end_lat	0
end_lng	0
member_casual	0

The task has been completed. Now, there are 4,803,698 rows and 11 columns remaining in the data set.

3.5 Adding New Columns

The fourth step in the data cleaning and manipulation process is to add three new columns to the data set:

- `month`: The month in which the trip was taken
- `day_of_week`: The day of the week on which the trip was taken
- `ride_length`: The duration of the trip in minutes

These columns were added using the `mutate()` function. It allows users to add new columns to a data set or to modify existing columns. There are currently 14 columns in the data set, as shown in Table 7. The addition of these new columns will make it possible to perform more detailed analysis of the data in the next section.

Table 7: A preview of Cyclistic's historical trip data

ride_id	98D355D9A9852BE9
rideable_type	classic_bike
started_at	2022-01-01 00:00:05
ended_at	2022-01-01 00:01:48
start_station_name	Michigan Ave & 8th St
end_station_name	Michigan Ave & 8th St
start_lat	41.87277
start_lng	-87.62398

end_lat	41.87277
end_lng	-87.62398
member_casual	casual
month	January
day_of_week	Saturday
ride_length	1.716667

3.6 Filter Unwanted Outliers

The final task in the data cleaning and manipulation process is to filter unwanted outliers. Outliers can be caused by errors in data entry, measurement, or other factors. They can distort the results of an analysis, so it is important to remove them before proceeding. In this case, all trip data that were less than 60 seconds were filtered out. These trips are likely to be false starts or users attempting to re-dock a bike to ensure its security. Trips that exceeded a period of 24 consecutive hours were also removed. This is because these trips are likely to be the result of the rider not returning the bike. This was accomplished by running the following R code:

```r
trip_data_v2 <- trip_data_v2[!( trip_data_v2$ride_length < 1 |
                                trip_data_v2$ride_length > 1440), ]
```

The presence of outliers in a data set can skew the results of statistical analyses. To ensure that the data set was free of outliers, the interquartile range (IQR) method was used to filter them out. The IQR is a measure of the variability of a data set, and it is calculated by subtracting the first quartile (Q1) from the third quartile (Q3). Outliers are defined as data points that are more than 1.5 times the IQR away from Q1 or Q3. The following R code was used to remove outliers from the data set:

```r
median_value <- median(trip_data_v2$ride_length)
iqr_value <- IQR(trip_data_v2$ride_length)
lower_limit <- median_value - 1.5 * iqr_value
upper_limit <- median_value + 1.5 * iqr_value

trip_data_v2 <- trip_data_v2[!( trip_data_v2$ride_length < lower_limit |
                                trip_data_v2$ride_length > upper_limit), ]
```

The trip data that were less than -8.54 minutes (technically 0 minutes) or greater than 29.91 minutes were deleted from the data set. This left 4,151,346 rows remaining after removing unwanted outliers. This completes the data cleaning and manipulation process, and the data is now ready for analysis.

4 Analysis and Key Findings

4.1 The Number of Rides

4.2 The Duration of Rides

4.3 The Location of Rides

4.4 Key Findings

Based on data analysis, :

-
-
-
-
-

5 Conclusion and Recommendations

5.1 Conclusion

Based on the data collected, it can be concluded that

5.2 Recommendations

The report makes the following recommendations:

1.
2.
3.