Gathering Data & Data Exploration

For this deliverable, I began by gathering only the yellow cab data from the s3 bucket and putting it into a bucket that I own. The data is separated into a csv file for each month of the years from 2009 to June 2021. I decided a simple way to analyze changes over time would be to collect the data for all months of 2009 and 2019 and otherwise only collect the csv for June of each year. So one analysis will be done on all data from June of each year and another analysis will be done on the change throughout the years of 2009 vs 2019 (since 2019 would be the most recent year with no affects from the pandemic). I uploaded the following csv files into an s3 bucket named bigdataprojectmadisonmelton:

* January – December 2009
* June 2010
* June 2011
* June 2012
* June 2013
* June 2014
* June 2015
* June 2016
* June 2017
* June 2018
* January – December 2019
* June 2020
* June 2021

While exploring the data I found that each csv file follows the below format. I selected the data types listed based on exploration of one of the csv files.

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Example value** | **Data Type** |
| VendorID | 1 | TINYINT |
| tpep\_pickup\_datetime | 6/1/2021 0:13 | datetime |
| tpep\_dropoff\_datetime | 6/1/2021 0:17 | datetime |
| passenger\_count | 1 | TINYINT |
| trip\_distance | 0.9 | FLOAT |
| RatecodeID | 1 | TINYINT |
| store\_and\_fwd\_flag | N | CHAR |
| PULocationID | 186 | SMALLINT |
| DOLocationID | 50 | SMALLINT |
| payment\_type | 1 | SMALLINT |
| fare\_amount | 5 | FLOAT |
| extra | 3 | FLOAT |
| tip\_amount | 0.5 | FLOAT |
| tolls\_amount | 2.2 | FLOAT |
| improvement\_surcharge | 0 | FLOAT |
| total\_amount | 0.3 | FLOAT |
| congestion\_surcharge | 11 | FLOAT |

Data Preparation Summary

* create s3 bucket in sandbox environment
* Separate CSV files into two folders and create the following tables:
  + June of years 2009-2021 (JuneData)
  + All months of 2009 and 2019 (2009vs2019)
* Use sql statements in athena to calculate the following for the June data:
  + Total passenger count for each year (monthly for JuneData)
  + average passenger count for each year (monthly for JuneData)
  + average fare for each year (monthly for JuneData)

The sandbox we were able to use for this project wouldn’t allow me to access my s3 bucket from Quicksight. I attempted to create a user who had permissions to do this, but creating a user was also not allowed. After trying a couple different sign-up options, I realized that the sandbox did not allow for the use of Quicksight in general. I decided the most effective way to create visualizations would be to collect the data I wanted to graph by querying in Athena and then list this data in a new excel file to put into tableau.

June Data

After loading all the data into the two folders, I used the below code to verify that the data from different years had correctly loaded in by replacing the ‘2021’ with different years:

SELECT \*

FROM "default"."junedata"

WHERE EXTRACT(YEAR FROM tpep\_pickup\_datetime)=2009

limit 10;

I then added a column to hold just the year from the full timestamp using the below code:

ALTER TABLE junedata ADD COLUMNS (year bigint)

I was going to then use the below statement to fill this column, however I got an error message and learned that update statements are not supported in Athena (<https://docs.aws.amazon.com/athena/latest/ug/other-notable-limitations.html>).

update junedata set year = EXTRACT(YEAR FROM tpep\_pickup\_datetime);

I then decided to just use the “EXTRACT(YEAR FROM tpep\_pickup\_datetime)=<year>” to gather the data I needed to put into a separate excel file. I also collected population data for New York city from a few sources to see how the number of rides shifted with population growth.

I used the below statement to get the average passenger count for each year:

SELECT AVG(passenger\_count)

FROM "yellowtaxidata"."junedata"

WHERE EXTRACT(YEAR FROM tpep\_pickup\_datetime)=2021;

I used the below statement to get the total passenger count for each year:

SELECT COUNT(\*)

FROM "yellowtaxidata"."junedata"

WHERE EXTRACT(YEAR FROM tpep\_pickup\_datetime)=2021;

I used the below statement to get the average fare for each year:

SELECT AVG(fare\_amount)

FROM "yellowtaxidata"."junedata"

WHERE EXTRACT(YEAR FROM tpep\_pickup\_datetime)=2021;

2009vs2019 Data

I used the below statement to get the total rides for each month in 2009 and 2019:

SELECT COUNT(\*)

FROM "yellowtaxidata"."2009vs2019"

WHERE EXTRACT(YEAR FROM tpep\_pickup\_datetime)=2019 AND EXTRACT(MONTH FROM tpep\_pickup\_datetime) = 1;

I used the below statement to get the average passenger count for each month in 2009 and 2019:

SELECT AVG(passenger\_count)

FROM "yellowtaxidata"."2009vs2019"

WHERE EXTRACT(YEAR FROM tpep\_pickup\_datetime)=2019 AND EXTRACT(MONTH FROM tpep\_pickup\_datetime) = 1;

I used the below statement to get the average fare amount for each month in 2009 and 2019:

SELECT AVG(fare\_amount)

FROM "yellowtaxidata"."2009vs2019"

WHERE EXTRACT(YEAR FROM tpep\_pickup\_datetime)=2019 AND EXTRACT(MONTH FROM tpep\_pickup\_datetime) = 1;

I also used the below statement to see what hours were the most popular for rides in 2009 vs 2019, however I did not use this data in the visualizations because they were the same.

SELECT COUNT(\*) as total

FROM "2009vs2019"

WHERE EXTRACT(YEAR FROM tpep\_pickup\_datetime) = 2019

GROUP BY EXTRACT(HOUR FROM tpep\_pickup\_datetime)

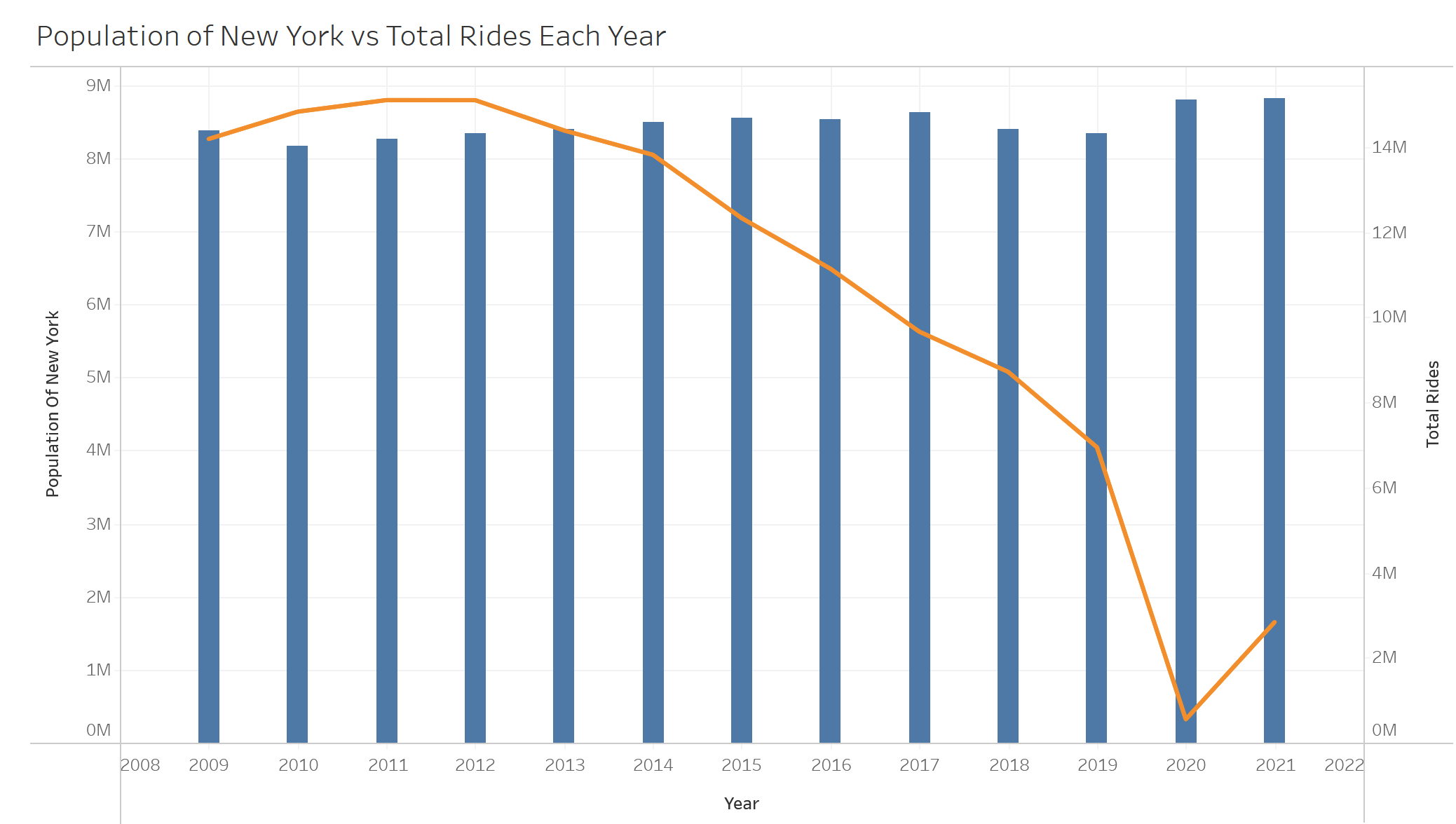
ORDER BY total DESC;

Visualizations in Tableau

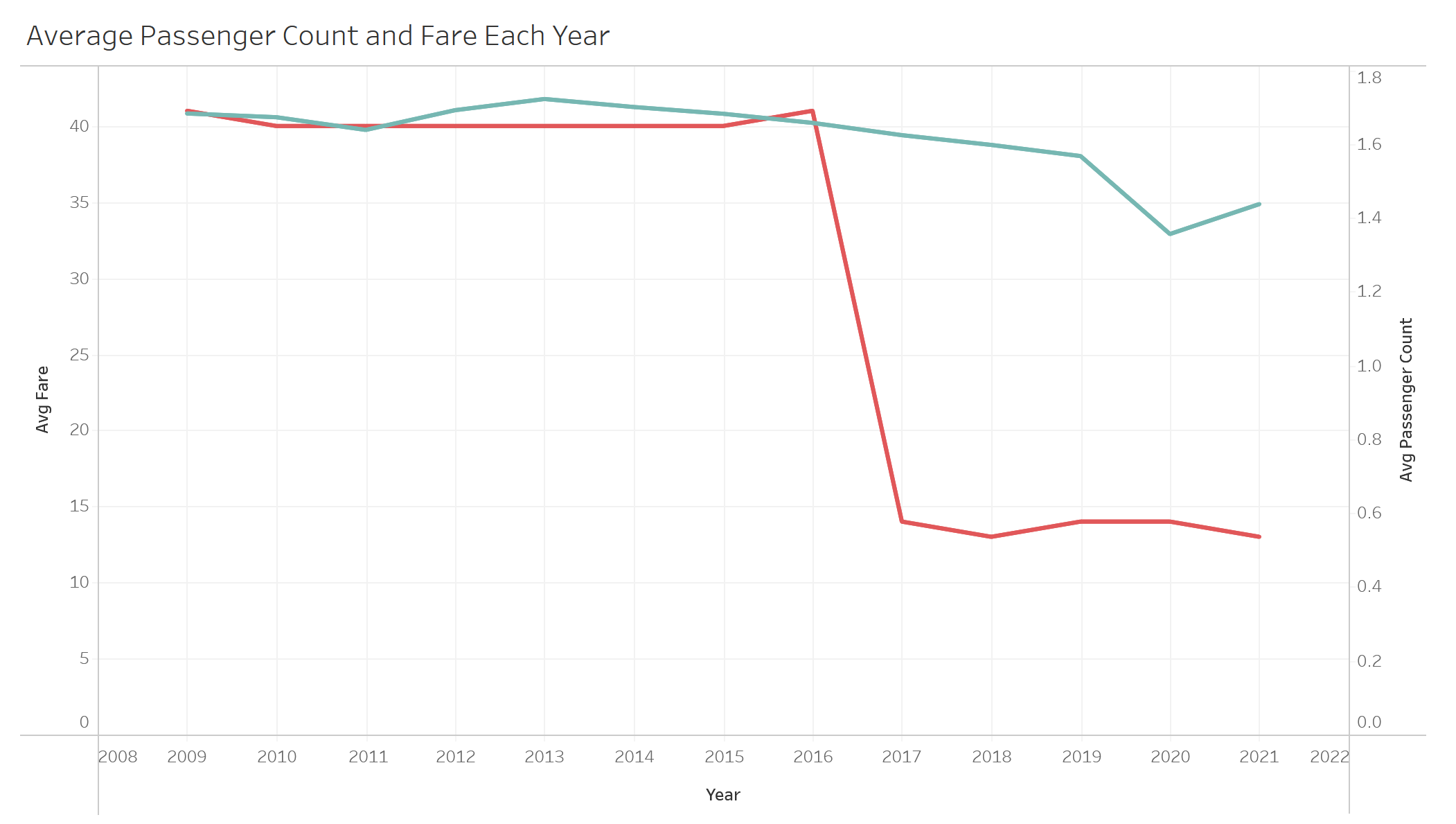
* Number of rides each June
* Number of rides each month in 2009 vs number of rides each month 2019
* average distance for each year
* average rate of dollar per mile

These are the two visualizations I created for the June data. I have placed pictures of them below but they are also available as an interactable graph at: <https://public.tableau.com/app/profile/madison.melton/viz/JuneData/JuneData?publish=yes>

Here we can see the population of New York on the left and in blue and the number of rides each year on the right and in orange. Uber came to New York in 2011 and has grown almost every year since. This is likely the cause for the steady drop off starting in 2012.

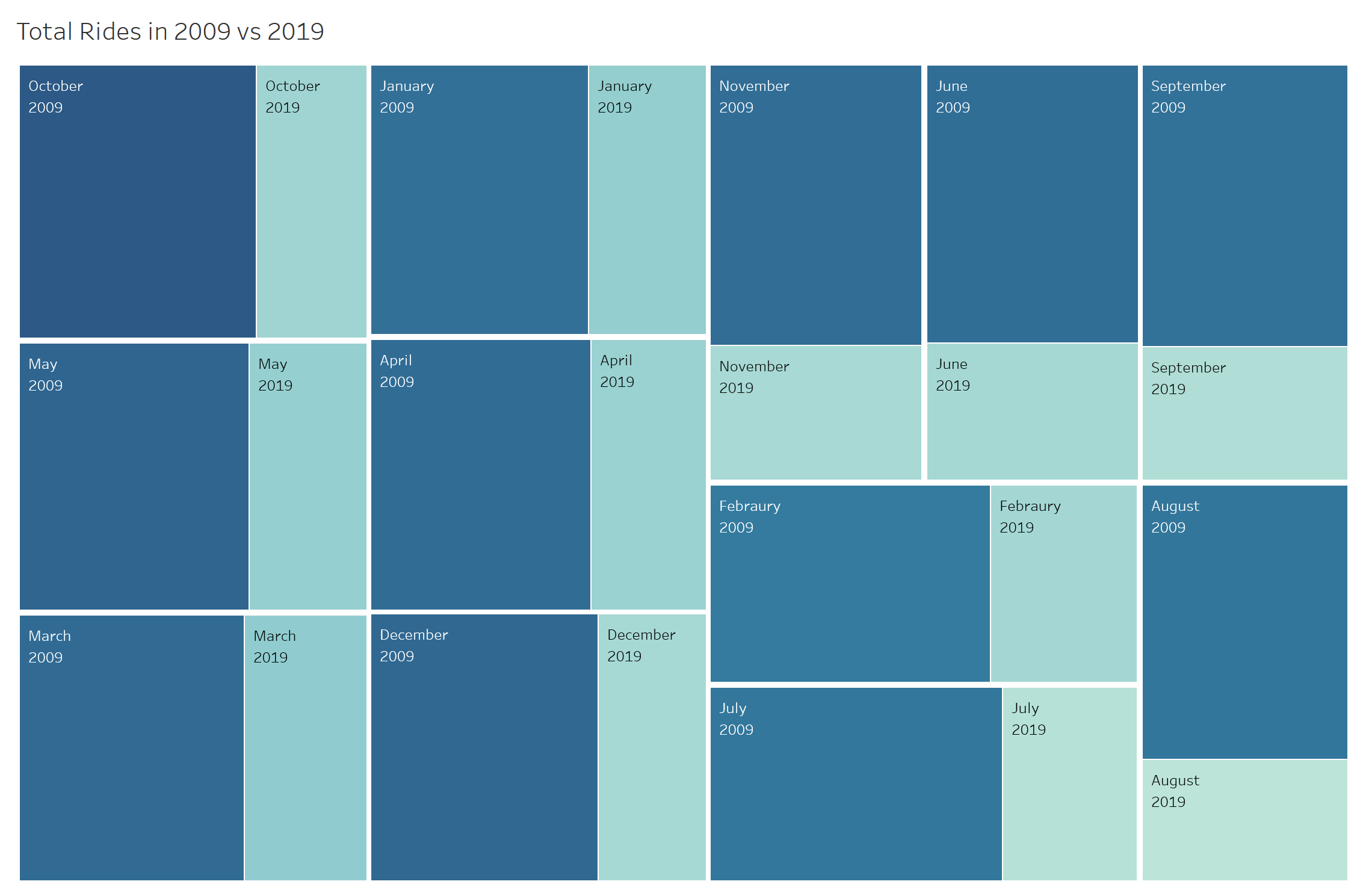


In this graph we can see the average fare and passenger count each year. I’m unsure why there would be such a drastic drop in the average fare, but it does seem that it was consistently around the $40s and dropped between 2016-2017 to be consistently around $13-15. This could be because once uber was available, if passengers were traveling a long distance they wanted to see the price ahead of time. So it’s possible people were more often calling cabs if they happened to be near one and were going a short distance.

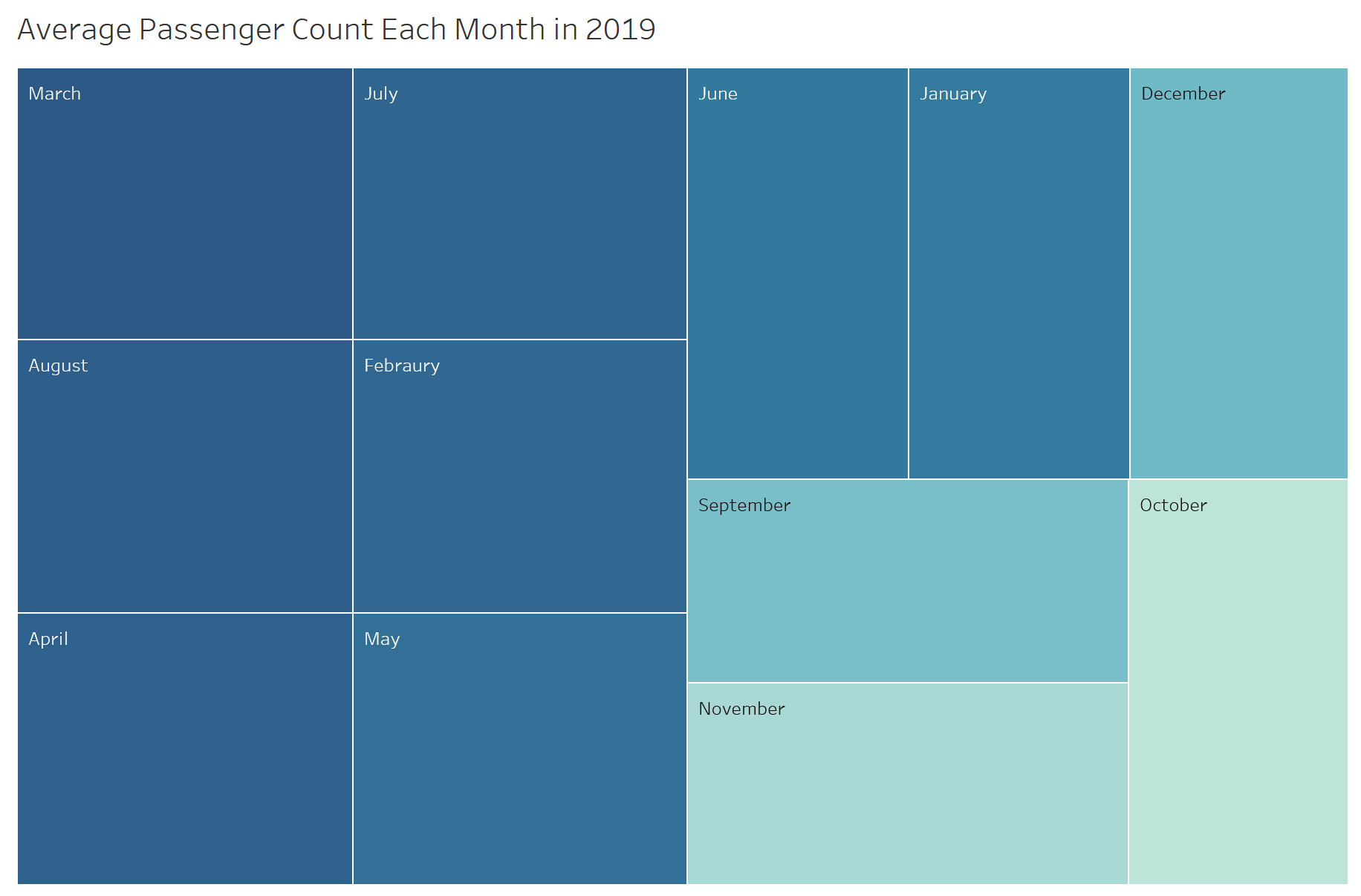


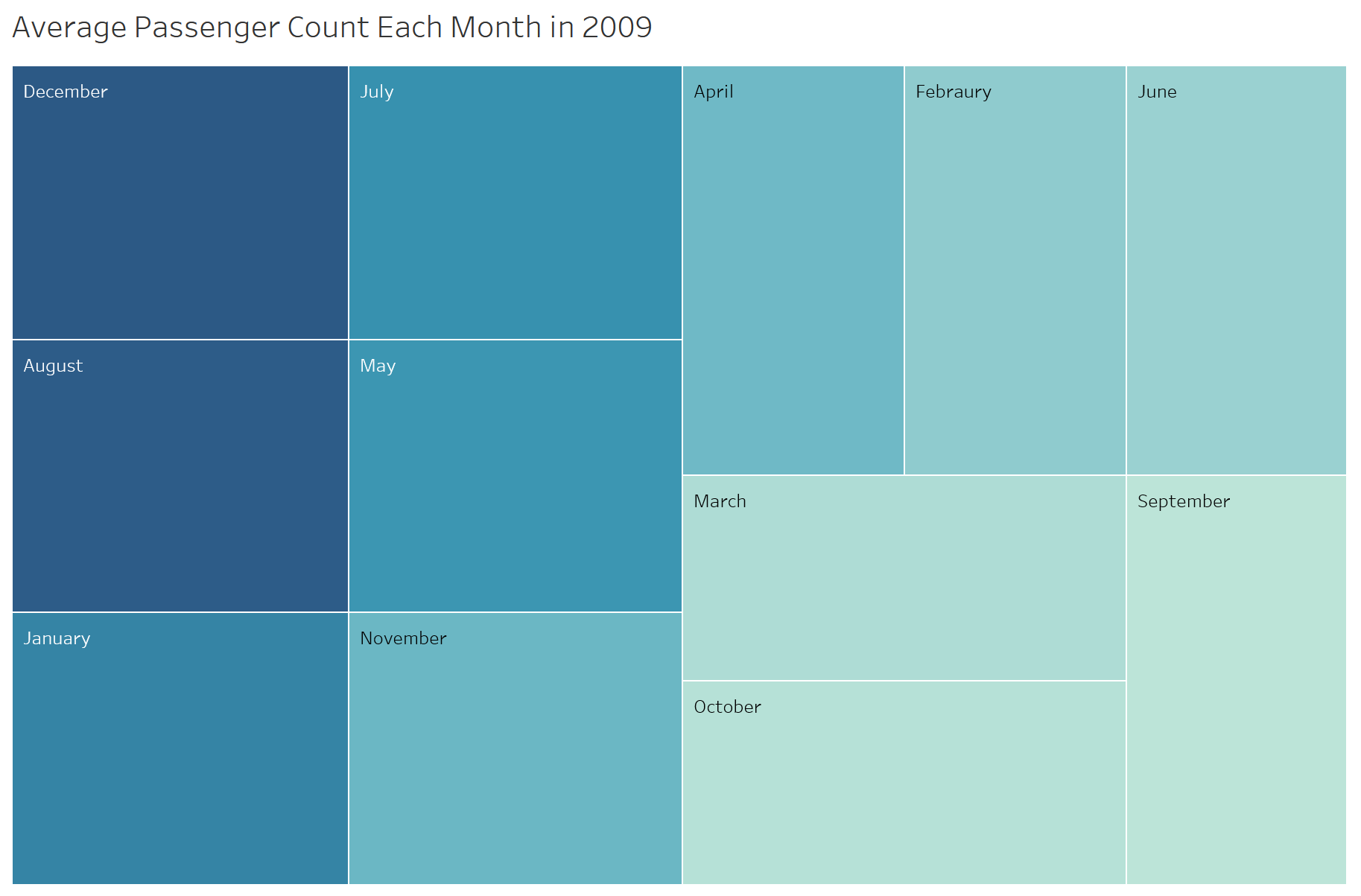
These are the three visualizations I created for the 2009vs2019 data. I have placed pictures of them below but they are also available as an interactable graph at: <https://public.tableau.com/app/profile/madison.melton/viz/2009vs2019Data/Dashboard2?publish=yes>

This graph depicts the total rides for each month of 2009 and 2019.



The graphs below depict the average passenger count for each month of 2009 and 2019. I found it interesting that the various months that had higher or lower passenger counts on average have changed a lot.





Bucket and table creation details

Bucket ARN: arn:aws:s3:::bigdataprojectmadisonmelton

**Bulk add info**

VendorID int,

tpep\_pickup\_datetime timestamp,

tpep\_dropoff\_datetime timestamp,

passenger\_count int,

trip\_distance int,

RatecodeID string,

store\_and\_fwd\_flag string,

PULocationID string,

DOLocationID string,

payment\_type string,

fare\_amount decimal,

extra decimal,

tip\_amount decimal,

tolls\_amount decimal,

improvement\_surcharge decimal,

total\_amount decimal,

congestion\_surcharge decimal

Important links

**Merging tables**

<https://www.w3schools.com/sql/sql_insert_into_select.asp>

Extract year from date/timestamp

<https://www.sqltutorial.org/sql-date-functions/how-to-extract-year-from-date-in-sql/>

New York city population data

<https://www.biggestuscities.com/city/new-york-new-york>

<https://www.census.gov/quickfacts/newyorkcitynewyork>

<https://www.populationu.com/cities/new-york-city-population>