Examining Consumer Ride Trends to Maximize Yellow Taxi Driver Profits in New York City

CONNOR MCCARTHY (cjm365@cornell.edu)

June 5, 2022

ABSTRACT

The rapid popularization of the gig economy, particularly in the ride-hailing industry, has put immense pressure on the long-standing New York City taxi system, limiting profits. We explore methods to combat this trend, utilizing data about taxi rides to reach conclusions and recommendations for drivers to employ. We analyze various types of rides distributions, identifying areas and times of high ride volume as well as sources of long rides. Using linear regression, we find that tip amounts can be largely explained by the fare amount, and, by plotting median trip duration, we identify a demographic of rider that tends to be more generous with their tips. After recommending that drivers maximize their time spent with riders in the car, we forecast ride volumes, identify a trend between weather and taxi popularity, and simulate various driver strategies.

CONTENTS

I. Introduction

The purpose of this recommendation report is to prescribe effective strategies in maximizing driver profits to yellow taxi drivers through analyzing consumer ride trends and transportation preferences in New York City. While the taxi business has enjoyed a storied history in New York City for nearly 100 years, it has been facing heightened competition in the past several years with the rapid adoption of ride-hailing apps, as illustrated in figure 1. By focusing on making recommendations to drivers of yellow cabs, our report will help these drivers compete with the newfound competition.

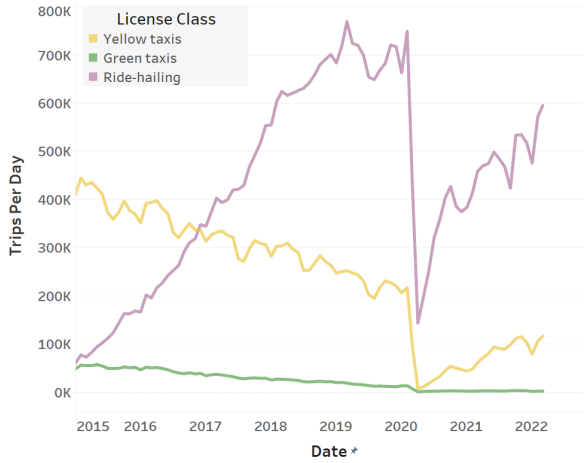


Figure 1: *Yellow (and green) taxis have been facing increased competition since the adoption of ride-hailing apps.*

The taxi industry in NYC is heavily regulated by the Taxi & Limousine Commission (TLC), who control the rate taxis can charge, supervise taxi trips, and, importantly, the total number of taxis allowed to operate in the city. The mechanism through which they do so, the medallion system, designates a certain number of physical medallions that are able to exist, and a valid medallion is required in order to operate the taxi [3]. The medallion market is highly competitive, and, in 2013, medallions were worth more than $1,000,000 each [4]. Taxi companies generally own or lease medallions, combining these medallions with taxi fleets, and leasing out these cars to taxi drivers. When driving the taxi however, drivers are able to retain 100% of all money collected — both fare and tip, less any expenses [2, 5].

As our team explores potential avenues of research for improving ride profitability for NYC yellow taxi drivers, we generated three questions to pave the way for potential solutions:

1. What trends in yellow taxi rides can be observed so as to better understand the demographic and behavior of riders?
2. Which factors of a taxi ride influence profitability the most?
3. What are tangible changes yellow taxi drivers can make to increase profitability for themselves?

In order to answer these questions, we've conducted extensive research, primarily utilizing information made public by the TLC of NYC. This dataset includes granular information about yellow taxi trips, green taxi trips, and less specific information about trips for High Volume For-Hire-Vehicles (HVFHV).[1] In addition, we've complemented this data with supporting data about weather conditions, as well as geospatial data that allows us to specify the location of each ride [6, 8]. All analysis conducted in this report was done after first filtering the dataset so as to remove outliers and incomplete records (appendix F), and, except where otherwise mentioned, was conducted based on trips that occurred during July of 2021.

II. Understanding the Data

In order to best understand the trends that underlie taxi rides in the city, we first investigate a number of descriptive visualizations about our data.

i. Geographic Distribution of Ride Volume

We approached visualizing the answer to the question of ride density by separating pickups and drop-offs by yellow (traditional) taxis, green (borough) taxis,[2] and ride-hailing app cars, as illustrated in figure 2.

We discovered that the number of pickups for yellow taxis was highest within the Manhattan region and the area surrounding JFK airport. Additionally, green taxis seemed to have the most popularity in upper Manhattan, with drop offs closer to Westchester, which, given their history, makes sense intuitively. Furthermore, ride-hailing app rides seemed to have rides spread out more

[1] HVFHV include Uber, Lyft, Via, and Juno

[2] green taxis were introduced in 2013 to supplement the classic yellow taxis and provide greater coverage to the greater NYC area [1]

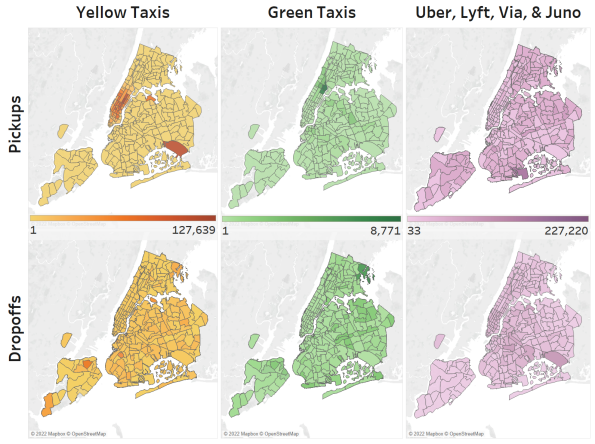


Figure 2: *Heat map of NYC rides, separated by yellow (traditional) taxis, green (borough) taxis, and ride-hailing app cars (Uber, Lyft, Via, Juno).*

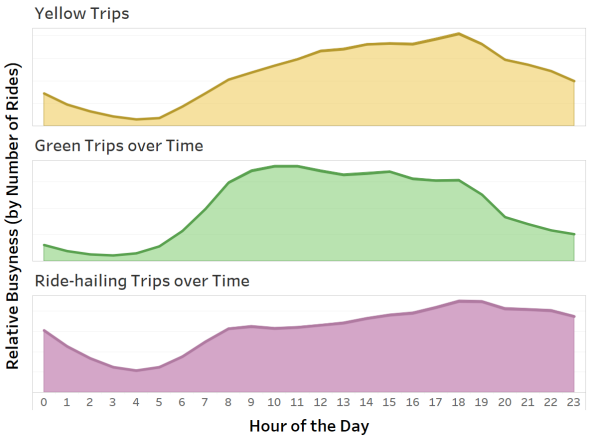


Figure 3: *Comparison of the number of trips throughout the day by ride type.*

evenly throughout the regions. These relationships are communicated through the heat map where the darker the region, the greater the number of rides from/to that region.

One interesting element of this analysis is that while green taxis are designed to provide for non-Manhattan traffic, being unable to pick up passengers below West 11th/East 96th Street, much of their traffic is still influenced by the crowds coming from uptown Manhattan [7].

In addition, while high volume for-hire vehicles have a tremendous amount of traffic, there exists no obvious trend among this traffic, indicating that it is not heavily influenced by a particular cause like commutes or tourism.

ii. Temporal Distribution of Ride Volume

An essential part of managing a small business centered around transportation, whether via NYC licensed taxis or via a ride-hailing service like Lyft or Uber, is maximizing the amount of time spent with passengers in the car. To do so, it is important to ensure that you are available during times of the day with a great deal of activity. As seen in figure 3, traffic begins to increase for the initial work commute around 6 am and, while ride-hailing apps tend to retain high traffic later into the night, traffic for NYC cabs, both yellow and green, tapers off around 7 pm.

As such, it might make sense for yellow cab drivers to target the times of day shown to have a high amount of traffic, especially the time period around 6 pm, where trip volume reaches its peak. Similarly, they may choose to avoid the late night/early morning shifts, realizing how difficult it will likely be to find enough riders to make it worth their while.

iii. Geographic Distribution of Trip Duration

An important aspect of any given ride is its duration, and, as discussed later on page 5, there are advantages and disadvantages to longer/shorter rides. In order to properly answer our 1st question and describe the behavior of taxi riders, we must understand how trip durations change with location.

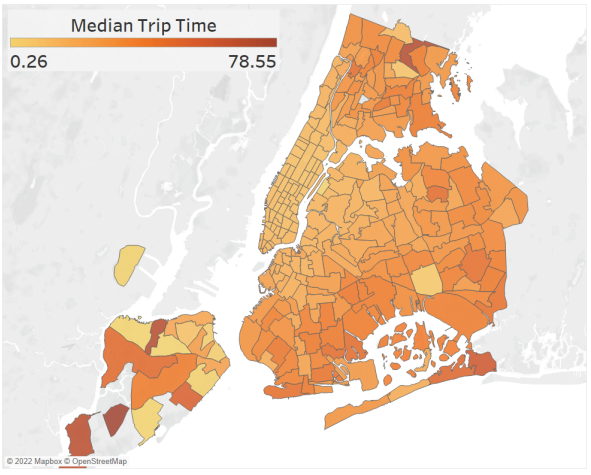


Figure 4: *Median trip time by pickup location for yellow cabs.*

Figure 4 depicts a map of the NYC area, with darker colors representing areas from where longer average trips

originate. Generally, the areas in Manhattan enjoy the shortest average trip duration, suggesting that many trips within Manhattan are relatively short and likely are not long-distance trips. As the distance from Manhattan increases, the median trip becomes longer and longer.

This information can potentially be explained by the lack of alternative public transportation in the boroughs relative to that in Manhattan. While taxi riders in Manhattan are often choosing to take a taxi for convenience's sake, those in the Bronx, Queens, Brooklyn, or Staten Island might be more likely to have no nearby subway/train station.

III. Maximizing Tips

As discussed earlier, the New York City taxi industry is generally well-established in its profit-structure: drivers lease[3] taxis from taxi companies for the duration of their shift, during which they are able to collect the entirety of the fares and tips they earn in that time period. Therefore, in order to maximize their profit, drivers should aim to increase either tip or fare amount indiscriminately. In this chapter we provide a thorough analysis of possible techniques to achieve the former.

i. Linear Regression Tip Predictions

When attempting to understand the tipping behavior of riders, it is important to do so in the context of the general American tipping behavior — it is common practice in America to leave a tip that is proportional to the total amount, often $16-20\%$.

Knowing this, we investigated the ability of the fare amount to predict the tip amount by training a linear model and fitting it to our data. As seen in figure 5, this model performs in a relatively passable fashion with a R-squared value of 0.624 and a p-value indicating that the fare amount does indeed have a significant effect (appendix A). This, generally speaking, means that 62% of the variation in tip amounts can be explained by the fare amount.

Another possible source of variation in tip amount is likely noise. Again, looking at figure 5, we can observe what seems to be some random noise or variation on both sides of the predictions. In the plot of the residuals, we

[3] due to the high cost of a medallion, it is rare for a taxi driver to own their own medallion

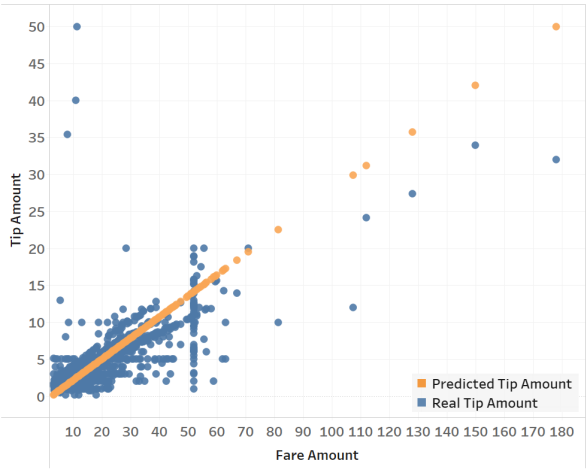


Figure 5: *Predicting Tip vs Fare, graphed for a $n = 2000$ subset of trips.*

notice the constant linear LOWESS line, indicating a constant residual mean, supporting our theory that there is a large random noise component (appendix Bi). However, this noise likely does not have constant variance, as observed by the changing slope of the LOWESS line when plotting the absolute value of the residuals (appendix Bii).

ii. Geographic Predictors of Large Tips

In an effort to better understand the demographic responsible for above-average tips, we began to investigate the geospatial attributes associated with large tips. In particular, we plotted the tip rate, calculated as tip amount divided by fare amount, by trip drop-off location and noticed that the median tip rate tended to be higher for trips ending in Manhattan than those elsewhere in the city (appendix C). As seen in figure 6, upon further inspection of Manhattan, this increase in tip rate seems to be primarily for trips terminating in midtown Manhattan rather than towards the north.

From this analysis, it becomes apparent that it might be more attractive, tip-wise, for drivers to target high-tipping riders heading to areas in midtown Manhattan rather than elsewhere, but such a strategy is easier said than done. To find the source of said riders, we attempted to map the average tip rate of drivers headed to Manhattan by their pickup location, but were unable to identify any trends (appendix D).

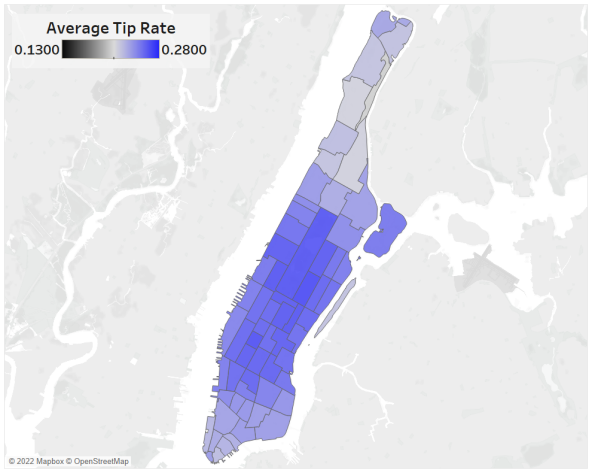


Figure 6: *Average tip rate by the geographic drop-off location within Manhattan.*

IV. Maximizing Fares

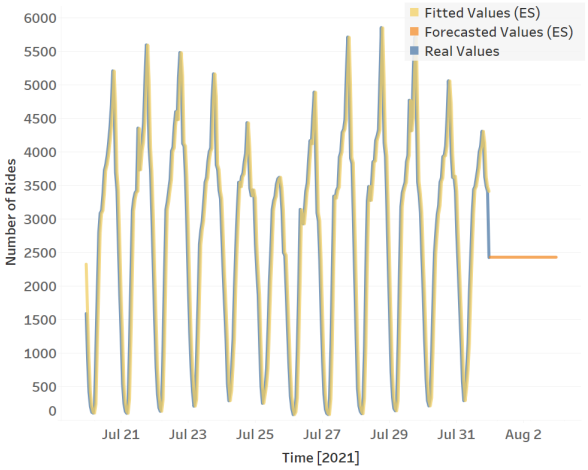
Similarly to how tip-maximizing strategies are considered, fare-maximizing techniques are a vital part to increasing the productivity, and therefore the profitability, of a taxi driver during their shifts. Due to the regulation put in place by the TLC, all taxi drivers in NYC must charge a uniform rate per amount driven. As such, the most effective way for a driver to increase their efficiency is to increase the amount of time they spend with riders in the car.

i. Forecasting Ride Volumes

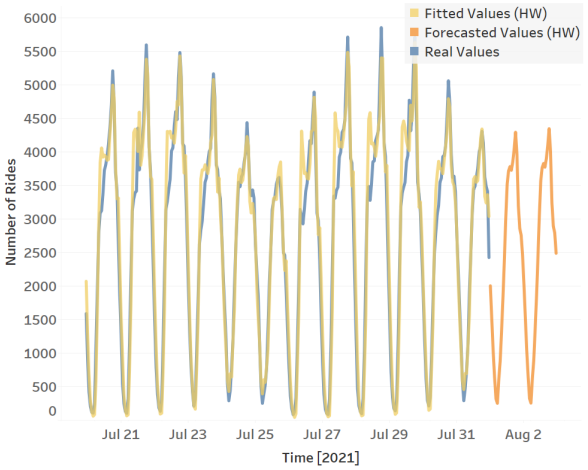
In large part due to their legal status as independent contractors rather than fulltime employees,[4] taxi drivers are often granted a great deal of autonomy regarding the hours they choose to work. As drivers often pay the taxi company to use their taxi for a certain amount of time, it is important for them to choose time periods during which they can maximize their productivity [2]. By finding times that enable them to conduct many trips, they are able to increase their total fare amount and, therefore, their profit.

To improve upon the visualization displayed in figure 3, we explored the use of forecasting models to fit and predict the ride volume over time with a greater deal of granularity. Both forecasting models, an exponential smoothing model[7i] and a Holt-Winters model,[7ii] allow

[4]the legal employment status of taxi drivers has been facing increased scrutiny in the last five years



(i) *Using exponential smoothing to fit and predict the number of taxi rides at a given time.*



(ii) *A Holt-Winters fit provides an improved fit and prediction for the number of taxi rides in a given time period.*

Figure 7: *Using two forecasting methods to fit the ride volume throughout July and predict volume for the first 48 hours of August.*

predictions into the future after being trained on historic data.

In the case of the exponential smoothing model, it assumes that the underlying data has a varying (noisy) mean and no underlying trend, and, as we can see in its prediction, it predicts a constant number of rides. In an effort to improve upon the quality of this prediction, we utilized the Holt-Winters model which similarly assumes a varying (noisy) mean, but this time also assumes the presence of an underlying trend. This time, the prediction for the following two rides appears much more precise, follow-

ing the general trend where rides have higher volume during daytime and rush hours than late at night.

In terms of the quality of fit for each forecast model, the sum of squared errors for the exponential smoothing was 1.957×10^8 in comparison to 1.105×10^8 for the Holt-Winters method. This difference in error indicates that the Holt-Winters method was able to achieve a more accurate fit, likely due to its ability to account for an underlying trend in the data, suggesting that the number of rides per day was slowly increasing throughout July.

ii. The Weather's Impact on Rider Behavior

One of the biggest advantages to being in NYC is the proximity of everything — many people are able to get around the city without needing a car, relying entirely on a combination of walking, biking, and public transportation. In fact, the average NYC yellow cab ride was only 3.2 miles, indicating that many taxi trips are likely within a walkable distance. On days with extreme temperatures or precipitation, people are likely less willing to walk and more likely to resort to hailing a cab.

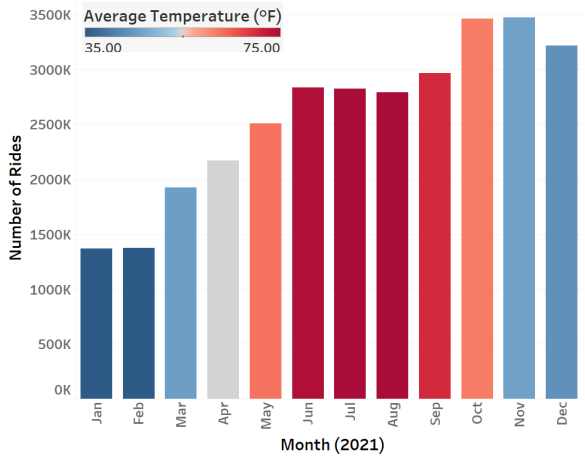


Figure 8: *The average number of trips per day for each month, colored by the month's average temperature.*

We investigated this phenomenon by comparing the volume of rides with the average temperature during a time period. Across 2021, no clear trend emerges, as trip volume steadily increases throughout the year, despite the cyclical weather pattern experienced in New York.[8] However, as seen in figure 9, hotter days do seem to consistently have more rides than their cooler counterparts. Upon fitting a linear regression model, we find that the temperature does have a statistically significant impact on total number of rides in a day, with each incremental degree of temperature increasing the total number of rides in a day by 964.7 on average (appendix E).

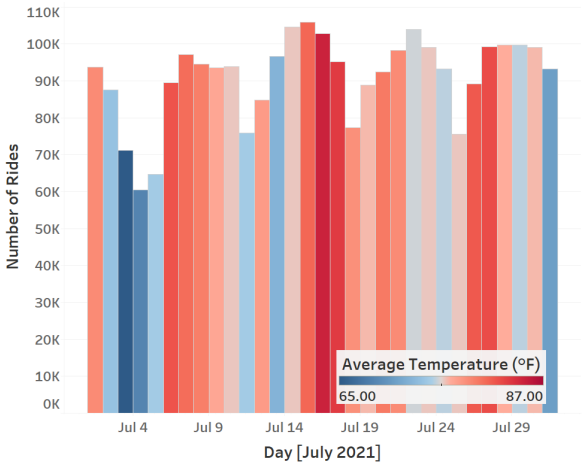
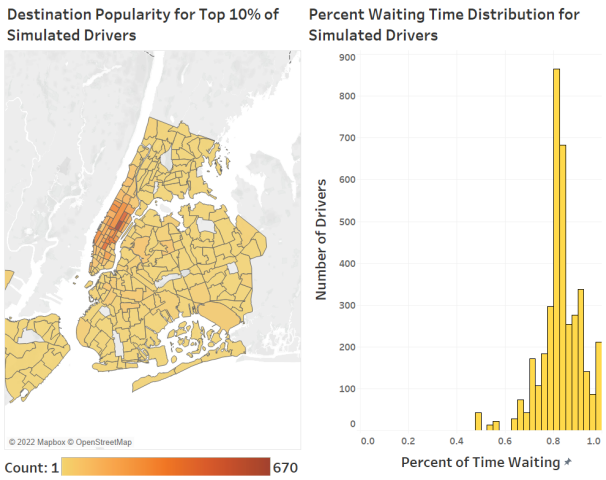


Figure 9: *The number of trips per day, colored by the average temperature during that day for the month of July.*

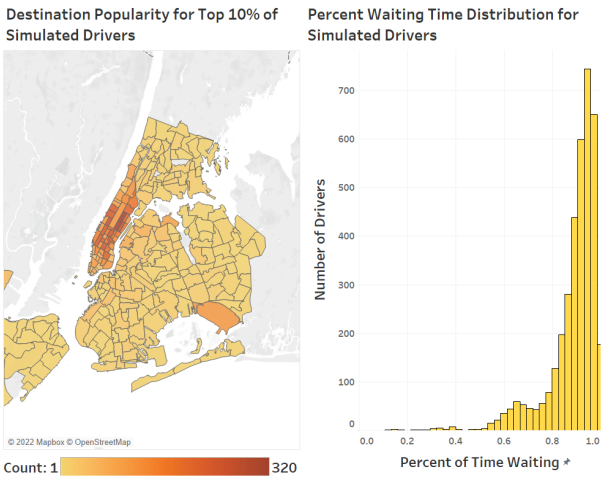
iii. Analyzing Driver Strategies

One of the pitfalls of driving a NYC yellow cab, especially relative to newer ride-hailing platforms like Uber and Lyft, is the lack of flexibility drivers have when it comes to dealing with demand spikes. Both Uber & Lyft employ a technique called "surge-pricing," in which temporary demand shocks or supply shortages are identified, and in order to ensure liquidity in the driver market, rates are temporarily increased. Taxis, because they are so heavily regulated by the Taxi and Limousine Commission, are not able to adjust the rate they offer in the same way. Therefore, there might seem to be less incentive for taxi drivers to find areas with surplus demand, provided they are able to keep their vehicle busy. Doing this, however, could be quite a challenging task. As a result, many drivers will sometimes be overly selective about the rides they select, in hopes that the "higher quality" riders they eventually take will lead them to situations where it is more convenient to find their next fare, thereby maximizing their time.

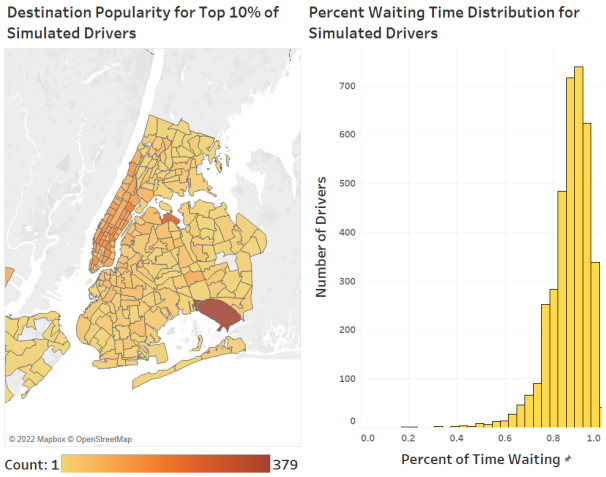
A common analysis technique for scenarios with uncertain outcomes is designing a Monte Carlo simulation. Such a technique involves the repeated simulation of a large quantity of situations, each involving a probabilistic



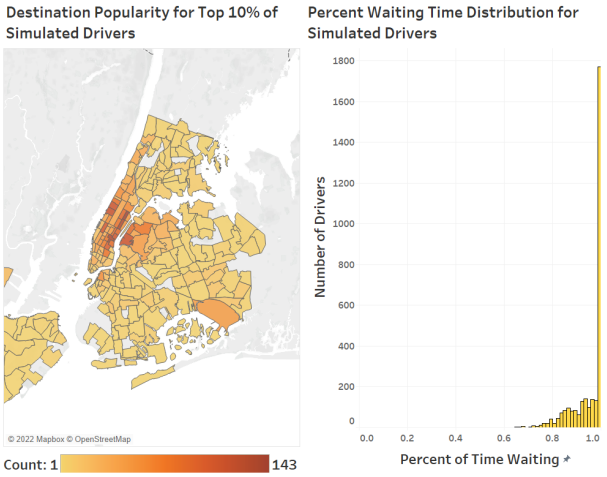
(i) *Driver Strategy: Accept First Possible Ride.*



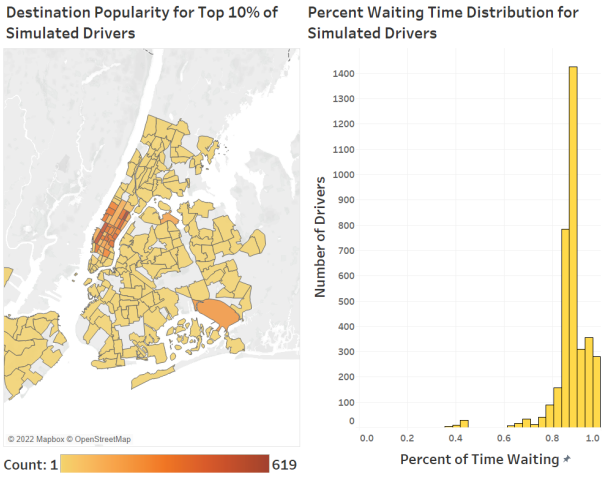
(ii) *Driver Strategy: Accept One of First Four Rides.*



(iii) *Driver Strategy: Choose Longer Rides.*



(iv) *Driver Strategy: Choose Shorter Rides.*



(v) *Driver Strategy: Prioritize Popular Destinations.*

Figure 10: *A simulation depicting popular destinations and wait times with varying strategies.*

event, and analyzing the results in the aggregate to draw conclusions. By simulating a NYC taxi driver's activity during a random day, and doing so many times for many drivers, it is possible to begin to draw conclusions about optimal strategies for drivers to employ. We have investigated a number of strategies that drivers might employ and plotted the amount of time that each driver spends without a rider, as well as the destinations that the most successful[5] tenth of drivers had traveled to.

As depicted in the various plots of figure 10, driver strategy can have a significant, possibly costly, impact on a taxi's profitability. Accepting the first rider encountered actually resulted in a relatively efficient allotment of time for the driver, indicating that the time gained with a rider in the car may have outweighed any negatives related to the quality of the rider's trip. Prioritizing riders headed to popular destinations did a similarly good job of minimizing driver wait time, indicating that the time spent waiting for a good ride might have been offset by a reduced amount of time to find the follow-up ride.

V. Final Recommendations

Through our analysis, we have observed a number of phenomena of interest to yellow taxi drivers, helping us to answer the questions that begun our investigation.

We identified trends with taxi rides and riders: the heightened demand for yellow taxis in the Manhattan area as well as near the JFK airport; the fluctuation of ride volume throughout the day which peaks during the evening rush hour; the propensity for rides originating from Manhattan to be shorter than those from the boroughs.

We've delved extensively into the components of profitability for a taxi driver, namely, total tip amount and total fare amount and have generated predictive models for each.

The understanding of these factors leads into our suggestions to drivers of changes to make in order to improve profitability. To maximize tips, we've suggested the prioritization of rides with higher fares and/or riders heading to midtown Manhattan, who tend to tip more generously. In order to maximize fare amount, we recommended prioritizing shifts during busy periods as forecast by our Holt-Winters model, shifts during hot days, and employing a rider selection strategy that decreases your waiting time.

For further steps to improve the profitablity of driving a yellow taxi, we recommend investigating the costs associated, such as fuel and car acquisition costs, as well as conducting further, more granular research as to rider behavior.

Overall, this analysis should provide yellow taxi drivers with a strong understanding of the NYC ride-hailing landscape, allowing them to contend more effectively with the increased competition from Uber, Lyft, and other ride-hailing apps.

[5] success was measured by the percentage of time a driver spent with a fare

VI. References

[1] Ali Bauman. *Longtime green taxi driver says NYC is doing little to help struggling drivers.* 2022. URL: https://www.cbsnews.com/newyork/news/green-taxi-drivers-new-york-city/.

[2] Sean Bryant. *How NYC's Yellow Cab Works and Makes Money.* 2015. URL: www.investopedia.com/articles/professionals/092515/how-nycs-yellow-cab-works-and-makes-money.asp.

[3] Lawrence Van Gelder. *Medallion Limits Stem from the 30's.* 1996. URL: www.nytimes.com/1996/05/11/nyregion/medallion-limits-stem-from-the-30-s.html.

[4] Winnie Hu. *Taxi Medallions, Once a Safe Investment, Now Drag Owners Into Debt.* 2017. URL: www.nytimes.com/2017/09/10/nyregion/new-york-taxi-medallions-uber.html.

[5] Chris Cumming Jeff Horwitz. *The Taxi Medallion System in New York and Other Cities Raises Fares, Impoverishes Drivers, and Hurts Passengers. So Why Can't We Get Rid of It?* 2012. URL: slate.com/business/2012/06/taxi-medallions-how-new-yorks-terrible-taxi-system-makes-fares-higher-and-drivers-poorer.html.

[6] National Centers for Environmental Information. *Global Summary of the Month.* 2021. URL: https://www.ncei.noaa.gov/access/search/data-search/global-summary-of-the-month.

[7] Taxi and Limousine Commission. *Green Cabs.* 2022. URL: https://www1.nyc.gov/site/tlc/businesses/green-cab.page.

[8] Taxi and Limousine Commission. *TLC Trip Record Data.* 2021. URL: www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page.

VII. Appendix

i. Supporting Figures and Visualizations

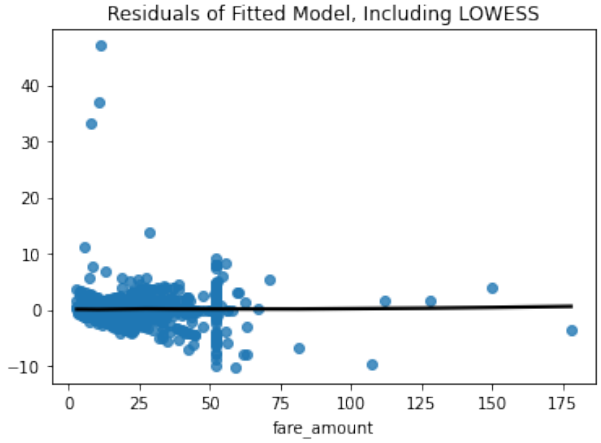
OLS Regression Results

Dep. Variable:	tip_amount	R-squared:	0.624
Model:	OLS	Adj. R-squared:	0.624
Method:	Least Squares	F-statistic:	3.199e+06
Date:	Sat, 14 May 2022	Prob (F-statistic):	0.00
Time:	11:30:54	Log-Likelihood:	-3.7636e+06
No. Observations:	1923812	AIC:	7.527e+06
Df Residuals:	1923810	BIC:	7.527e+06
Df Model:	1		
Covariance Type:	nonrobust		

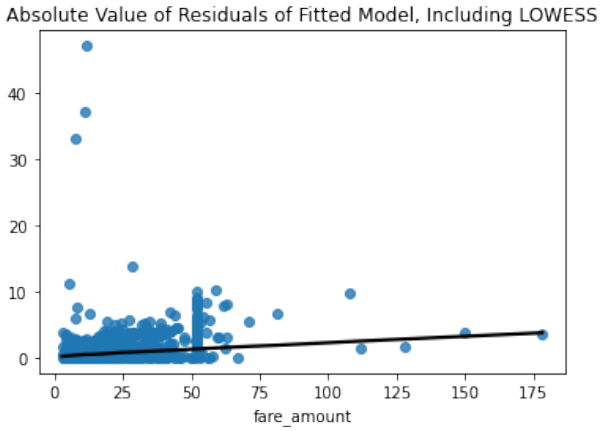
	coef	std err	t	P>|t|	[0.025	0.975]
const	0.7073	0.002	376.640	0.000	0.704	0.711
fare_amount	0.1957	0.000	1788.587	0.000	0.195	0.196

Omnibus:	5006248.891	Durbin-Watson:	1.992
Prob(Omnibus):	0.000	Jarque-Bera (JB):	1133155058039.630
Skew:	29.214	Prob(JB):	0.00
Kurtosis:	3762.383	Cond. No.	26.2

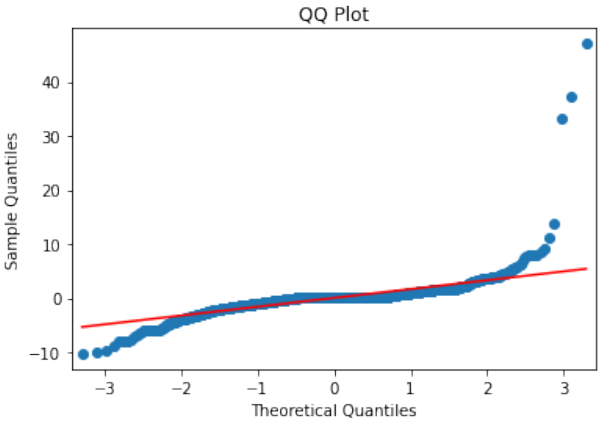
Figure A: *The summary of our regression model when trained on fare amount for the full (filtered) dataset, attempting to predict tip amount.*



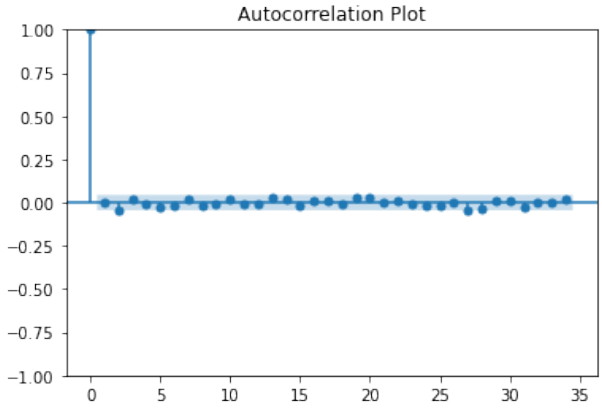
(i) *Residuals*



(ii) *Absolute Value of Residuals*



(iii) *QQ Plot*



(iv) *Autocorrelation Plot*

Figure B: *Verifying assumptions for linear regression on tip amount.*

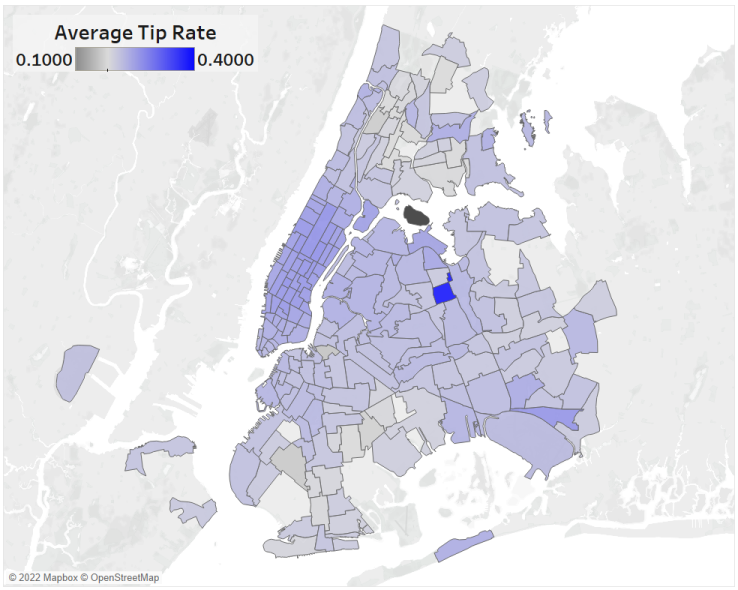


Figure C: *Tip rate throughout NYC based on trip departure location.*

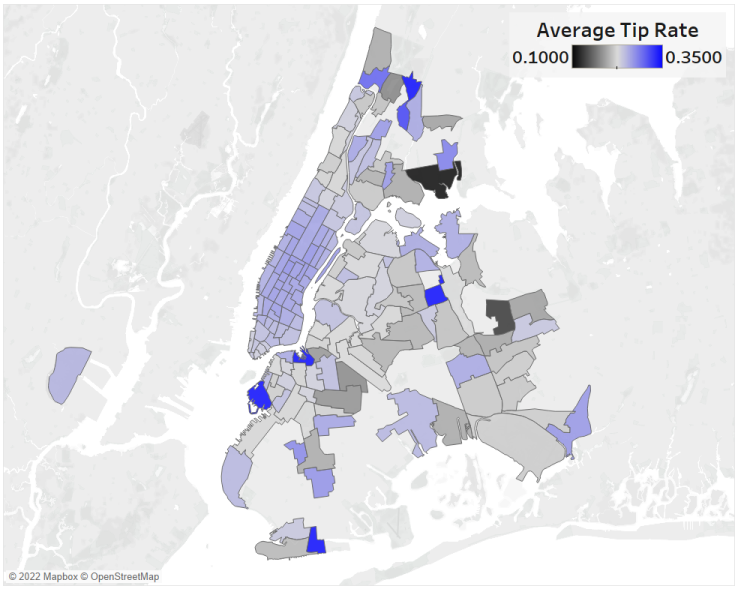


Figure D: *Tip rate of trips heading to Manhattan, plotted by trip departure location.*

OLS Regression Results

Dep. Variable:	count	R-squared:	0.275
Model:	OLS	Adj. R-squared:	0.250
Method:	Least Squares	F-statistic:	11.02
Date:	Sun, 15 May 2022	Prob (F-statistic):	0.00244
Time:	16:04:08	Log-Likelihood:	-320.29
No. Observations:	31	AIC:	644.6
Df Residuals:	29	BIC:	647.5
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-1.254e+04	2.25e+04	-0.557	0.582	-5.86e+04	3.35e+04
TAVG	964.7335	290.556	3.320	0.002	370.479	1558.988

Omnibus:	0.874	Durbin-Watson:	0.779
Prob(Omnibus):	0.646	Jarque-Bera (JB):	0.902
Skew:	-0.340	Prob(JB):	0.637
Kurtosis:	2.513	Cond. No.	1.26e+03

Figure E: *The summary of the linear regression model relating average temperature with total number of rides.*

ii. Relevant Code Excerpts

```
col_data_types = {
    'VendorID' : int,
    # 'tpep_pickup_datetime' : str,
    # 'tpep_dropoff_datetime' : str,
    'passenger_count' : int,
    'trip_distance' : float,
    'RatecodeID' : int,
    'store_and_fwd_flag' : str,
    'PULocationID' : int,
    'DOLocationID' : int,
    'payment_type' : int,
    'fare_amount' : float,
    'extra' : float,
    'mta_tax' : float,
    'tip_amount' : float,
    'tolls_amount' : float,
    'improvement_surcharge' : float,
    'total_amount' : float,
    'congestion_surcharge' : float
}
num_locations = 265

df_yellow.dropna(inplace= True)
df_yellow = df_yellow.astype(col_data_types)

df_yellow = df_yellow[df_yellow['payment_type']==1.0]
df_yellow = df_yellow[df_yellow['tpep_dropoff_datetime'] - df_yellow['tpep_pickup_datetime'] > datetime.
    timedelta(minutes=1)]
df_yellow = df_yellow[df_yellow['tpep_dropoff_datetime'] - df_yellow['tpep_pickup_datetime'] < datetime.
    timedelta(hours=12)]
df_yellow = df_yellow[df_yellow['total_amount'] < 1500]
df_yellow = df_yellow[df_yellow['total_amount'] > 1]
df_yellow = df_yellow[df_yellow['PULocationID'] <= num_locations]
df_yellow = df_yellow[df_yellow['DOLocationID'] <= num_locations]
df_yellow = df_yellow[df_yellow['tip_amount'] > 0.10]
df_yellow['duration'] = df_yellow['tpep_dropoff_datetime'] - df_yellow['tpep_pickup_datetime']
df_yellow = df_yellow[df_yellow['tpep_dropoff_datetime'].dt.year == 2021]
df_yellow = df_yellow[df_yellow['tpep_dropoff_datetime'].dt.month == 7]

df_yellow.head()
```

Figure F: *Python code used to filter the main yellow taxi trip dataset, removing outliers and incomplete records.*

```python
def location_popularity_normal():
    locations = df_yellow.groupby(['PULocationID']).size()
    num_locations = locations.shape
    total_pickups = locations.sum()
    return (locations * num_locations) / total_pickups

location_popularity_normals = location_popularity_normal()


def choose_next_ride_always(row, idx):
    if pd.isna(row).any():
        return pd.Series([pd.NA, pd.NA, pd.NA])

    trips_in_location = df_yellow[df_yellow['PULocationID'] == row[f'{idx-1}_locationID']]
    future_trips_in_location = trips_in_location[trips_in_location['tpep_pickup_datetime'] > row[f'{idx-1}_dropoff_time']]

    num_trips, _ = future_trips_in_location.shape
    if num_trips == 0:
        return pd.Series([pd.NA, pd.NA, pd.NA])

    result= pd.Series([future_trips_in_location['tpep_pickup_datetime'].iloc[0],
                       future_trips_in_location['tpep_dropoff_datetime'].iloc[0],
                       future_trips_in_location['DOLocationID'].iloc[0]],
                      )
    return result

def choose_next_ride_probability(row, idx):
    if pd.isna(row).any():
        return pd.Series([pd.NA, pd.NA, pd.NA])

    trips_in_location = df_yellow[df_yellow['PULocationID'] == row[f'{idx-1}_locationID']]
    future_trips_in_location = trips_in_location[trips_in_location['tpep_pickup_datetime'] > row[f'{idx-1}_dropoff_time']]

    ride_to_choose = random.randint(0, 3)

    num_trips, _ = future_trips_in_location.shape
    if num_trips <= ride_to_choose:
        return pd.Series([pd.NA, pd.NA, pd.NA])


    result= pd.Series([future_trips_in_location['tpep_pickup_datetime'].iloc[ride_to_choose],
                       future_trips_in_location['tpep_dropoff_datetime'].iloc[ride_to_choose],
                       future_trips_in_location['DOLocationID'].iloc[ride_to_choose]],
                      )
    return result

def choose_ride_by_long_length(row, idx):
    if pd.isna(row).any():
        return pd.Series([pd.NA, pd.NA, pd.NA])

    trips_in_location = df_yellow[df_yellow['PULocationID'] == row[f'{idx-1}_locationID']]
    future_trips_in_location = trips_in_location[trips_in_location['tpep_pickup_datetime'] > row[f'{idx-1}_dropoff_time']]

    length_threshold = datetime.timedelta(minutes= np.random.uniform(0.75, 1.25) * 20)
    future_possible_trips = future_trips_in_location[future_trips_in_location['tpep_dropoff_datetime'] - future_trips_in_location['tpep_pickup_datetime'] > length_threshold]

    num_trips, _ = future_possible_trips.shape
    if num_trips == 0:
        return pd.Series([pd.NA, pd.NA, pd.NA])


    result= pd.Series([future_possible_trips['tpep_pickup_datetime'].iloc[0],
                       future_possible_trips['tpep_dropoff_datetime'].iloc[0],
                       future_possible_trips['DOLocationID'].iloc[0]],
                      )
    return result
```

Figure G: *Python code responsible for implementing the choose driver strategies to simulate (part 1).*

```python
def choose_ride_by_short_length(row, idx):
    if pd.isna(row).any():
        return pd.Series([pd.NA, pd.NA, pd.NA])

    trips_in_location = df_yellow[df_yellow['PULocationID'] == row[f'{idx-1}_locationID']]
    future_trips_in_location = trips_in_location[trips_in_location['tpep_pickup_datetime'] > row[f'{idx-1}_dropoff_time']]

    length_threshold = datetime.timedelta(minutes= np.random.uniform(0.75, 1.25) * 5)
    future_possible_trips = future_trips_in_location[future_trips_in_location['tpep_dropoff_datetime'] - future_trips_in_location['tpep_pickup_datetime'] < length_threshold]

    num_trips, _ = future_possible_trips.shape
    if num_trips == 0:
        return pd.Series([pd.NA, pd.NA, pd.NA])


    result= pd.Series([future_possible_trips['tpep_pickup_datetime'].iloc[0],
                       future_possible_trips['tpep_dropoff_datetime'].iloc[0],
                       future_possible_trips['DOLocationID'].iloc[0]],
                    )
    return result

def choose_ride_by_dest_popularity(row, idx):
    if pd.isna(row).any():
        return pd.Series([pd.NA, pd.NA, pd.NA])

    trips_in_location = df_yellow[df_yellow['PULocationID'] == row[f'{idx-1}_locationID']]
    future_trips_in_location = trips_in_location[trips_in_location['tpep_pickup_datetime'] > row[f'{idx-1}_dropoff_time']]

    popularity_thresh = np.random.uniform(2, 3)
    future_possible_trips = future_trips_in_location[future_trips_in_location['DOLocationID'].map(location_popularity_normals) > popularity_thresh]

    num_trips, _ = future_possible_trips.shape
    if num_trips == 0:
        return pd.Series([pd.NA, pd.NA, pd.NA])


    result= pd.Series([future_possible_trips['tpep_pickup_datetime'].iloc[0],
                       future_possible_trips['tpep_dropoff_datetime'].iloc[0],
                       future_possible_trips['DOLocationID'].iloc[0]],
                    )
    return result

def simulate(drivers, strategy, number_iterations = 10):
    experiment_start_time = datetime.datetime(2021, 7, random.randint(1, 16), random.randint(0, 23), random.randint(0,59), 0)
    drivers['0_dropoff_time'] = experiment_start_time
    drivers = drivers.iloc[:, ::-1]

    for i in range(1, number_iterations + 1):
        print(f"Simulating iteration {i} / {number_iterations}...")
        drivers[[f'{i}_pickup_time', f'{i}_dropoff_time', f'{i}_locationID']] = drivers.apply(lambda row: strategy(row, i), axis= 1, result_type='expand')

    return drivers
```

Figure H: *Python code responsible for implementing the choose driver strategies to simulate (part 2).*

```python
def calc_waiting_time(row, i):
    if pd.isna(row[f"{i}_pickup_time"]) or pd.isna(row[f"{i-1}_dropoff_time"]):
        return [pd.NA]
    result = pd.Series(row[f"{i}_pickup_time"] - row[f"{i-1}_dropoff_time"])
    return result


def calc_driving_time(row, i):
    if pd.isna(row[f"{i}_dropoff_time"]) or pd.isna(row[f"{i}_pickup_time"]):
        return [pd.NA]
    result = pd.Series(row[f"{i}_dropoff_time"] - row[f"{i}_pickup_time"])
    return result


def calc_usage_rates(sim):
    print(f"in calc_usage_rates {sim.shape=}")
    n_drivers, n_iters = sim.shape
    n_iters = (n_iters - 2) // 3
    times = pd.DataFrame()

    free_time = pd.DataFrame()
    for i in range(1, n_iters):
        free_time[f"driver_time_waiting_at_{i}"] = sim.apply(
            lambda row: calc_waiting_time(row, i), axis=1, result_type="expand"
        )

    free_time = free_time.fillna(datetime.timedelta(0))
    times["total_time_waiting"] = free_time.apply(
        lambda row: sum(row, datetime.timedelta(0)), axis=1
    )
    times.reset_index(drop=True, inplace=True)

    def last_valid(df):
        df_times = df.iloc[::3]
        if df_times.last_valid_index() is None:
            return np.nan
        else:
            return df_times[df_times.last_valid_index()]

    end_of_driving = pd.to_datetime(sim.apply(last_valid, axis=1))
    start_of_driving = sim["0_dropoff_time"]
    times["total_incar_time"] = end_of_driving - start_of_driving
    times.reset_index(drop=True, inplace=True)

    result = times["total_time_waiting"].div(times["total_incar_time"])
    return result
```

Figure I: *Python code that calculates the relevant statistics for the simulated drivers, exporting them to a csv file for later reference (part 1).*

```python
def simulation_to_final_data(sim, fname):
    usage_rates = calc_usage_rates(sim)
    usage_rates = usage_rates.replace([np.inf, -np.inf], np.nan).dropna()
    usage_rates.columns = ["percent time driving"]
    usage_rates.index.name = "driver"
    # plotting usage rates
    plt.hist(usage_rates, bins=100)
    plt.axvline(x=usage_rates.mean(), c="r", linestyle="--")
    plt.show()
    # save usage rate to csv
    usage_rates.to_csv(
        f"montecarlo_2/{fname}_usage_rates.csv", header=["percent time driving"]
    )
    print(f"Saved file: {fname}_usage_rates.csv")

    num_top_ten_percent = usage_rates.shape[0] // 10
    top_ten_usage_rates_ind = np.argsort(usage_rates, axis=1).iloc[:num_top_ten_percent]
    top_ten_usage_rates = sim.iloc[top_ten_usage_rates_ind]
    locationIDs = top_ten_usage_rates.iloc[:, 1::3]
    locationIDs = pd.DataFrame(locationIDs.to_numpy().flatten(), columns=["values"])
    location_counts = locationIDs.groupby(["values"]).size()
    location_counts.columns = ["count"]
    location_counts.index.name = "locationID"
    # plotting location counts of top 10 drivers
    plt.plot(location_counts)
    plt.xlabel(f"{num_top_ten_percent} drivers in 10th percentile")
    plt.show()
    # save counts to csv
    location_counts.to_csv(
        f"montecarlo_2/{fname}_location_counts.csv", header=["count"]
    )
    print(f"Saved file: {fname}_location_counts.csv")
```

Figure J: *Python code that calculates the relevant statistics for the simulated drivers, exporting them to a csv file for later reference (part 2).*