Flatten the Curve

A.I Project
Joh Eun Kang, Dinesh Sreekanthan

Introduction:

In the spring of 2020, the world has experienced an unprecedented event in modern historical times.
A novel virus called “Coronavirus” killed tens of thousands of people around the world. Due to the pandemic from the COVID19, countless people lost their jobs.

It has been around for three months, but the virus is only getting stronger. Because of this powerful contagion, people expect it to take a long time to get back to the normal life we lived in. Some also say it would be impossible to return to our previous lives.

So, we must make changes in many ways for this "new normal life". It is necessary to come up with ways to maintain the distance between people and people and make efforts to prevent the population from being concentrated in one place as much as possible.

New York is a densely populated area, and it is a city with more public transportation than other states. Therefore, it is difficult to maintain a certain distance between people, and it is easy to find yourself in heavily crowded areas.

Our project is a taxi reservation system for the "new normal". Until now, people have been booking taxis freely at any time and place. However, this reservation system is difficult to implement in our current situation. In order to combat this issue, we enforce some restrictions on this reservation system.

Based on the specified threshold, the reservation system will approve if the number of people at the location entered by the user is less than the threshold. Otherwise, the reservation system will reject it.

Data:

https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page
2019 yellow cap data

Data length: 6567788

Data Features:

- Original data: VendorID, tpep_pickup_datetime, tpep_dropoff_datetime, passenger_count, trip_distance, RatecodeID',store_and_fwd_flag',PULocationID','DOLocationID','payment_type', 'fare_amount','extra', 'mta_tax','tip_amount', 'tolls_amount', 'improvement_surcharge','total_amount', 'congestion_surcharge'
- Used Features: 'tpep_pickup_datetime','tpep_dropoff_datetime', 'passenger_count','PULocationID', 'DOLocationID', 'Type'

Assumptions:

We set some assumptions for the shorter running time.

1. We are only using 2019 September data
2. Only using data of Manhattan area location number 43 (central park): the central park is one of the locations that are always dense with people.
3. Time window (time threshold): 2 hours
4. Reject reservation if there are more people than threshold

Limitation:

This project is for the idea of the system. Since we are using a small amount of data compared to real-world data, there are some limitations for using this project in practice.

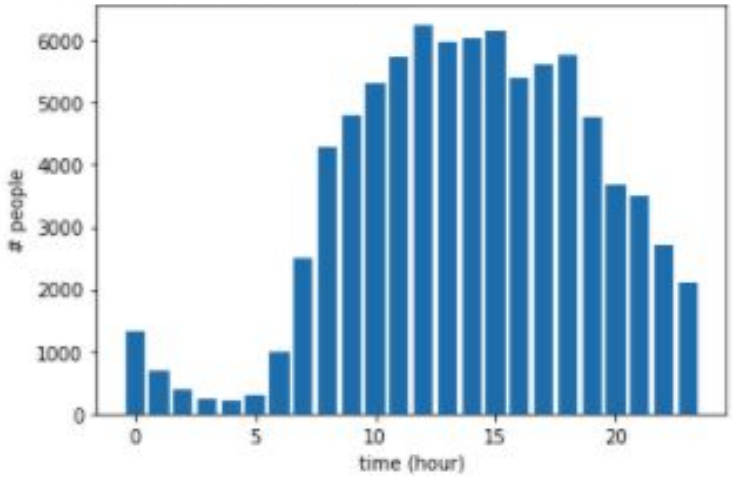
1. Not a real-time system
2. Not updating new data
3. Data collected in different situation (not a pandemic)

Experiment:

Part 1 (Data processing):

In order to implement the first part of the project, we first remove some redundant features and add “Type” to make sure our data is only using the yellow-cap. Then, we separate drop-off time to hours (h), minute (m), and seconds (s) to satisfy a specific time window (hour-wise).

Then, we take the data that satisfies our specific location ID (43, central park). From this new data, we count the number of passengers with respect to hours (0-23, 24 hours) so that we have a visualization of time vs number of passengers.



In September 2019, the central park was dense during 9am to 8pm.

Part 2 (Decision Step):

In this part, we take user's input such as reservation date (%Y-%m-%d), drop off location id, and the time (ex. 12:30). Since we assumed that we set a drop off location id as 43 central park Manhattan, the user input should be 43. We set the threshold of the number of people in this location by calculating the average over the plot.

In order to help the user, we provide a table showing the dropoff location ID to the borough and zone. We source this data from **https://s3.amazonaws.com/nyc-tlc/misc/taxi+_zone_lookup.csv** Since we are only dealing with Yellow zones, we filter out only the Yellow zone areas and display it to the user in a readable manner to make it easier for them to choose a location ID. In our test cases, we use location ID 43, which corresponds to Central Park in Manhattan.

	LocationID	Borough	Zone
3	4	Manhattan	Alphabet City
11	12	Manhattan	Battery Park
12	13	Manhattan	Battery Park City
23	24	Manhattan	Bloomingdale
42	43	Manhattan	Central Park
44	45	Manhattan	Chinatown
47	48	Manhattan	Clinton East
49	50	Manhattan	Clinton West
67	68	Manhattan	East Chelsea
78	79	Manhattan	East Village
86	87	Manhattan	Financial District North
87	88	Manhattan	Financial District South
89	90	Manhattan	Flatiron
99	100	Manhattan	Garment District
102	103	Manhattan	Governor's Island/Ellis Island/Liberty Island
103	104	Manhattan	Governor's Island/Ellis Island/Liberty Island
104	105	Manhattan	Governor's Island/Ellis Island/Liberty Island
106	107	Manhattan	Gramercy
112	113	Manhattan	Greenwich Village North
113	114	Manhattan	Greenwich Village South
124	125	Manhattan	Hudson Sq
136	137	Manhattan	Kips Bay
139	140	Manhattan	Lenox Hill East
140	141	Manhattan	Lenox Hill West

After the system gets the user's input, it gives the current population of the location (43) and maximum allowed safe capacity (threshold), and rejection/ acceptation. If the population at the user's input time is less than the system threshold, the system accepts the reservation. However, if it exceeds the threshold, the system will reject the reservation, and give the user a list of available reservation time.

Result:

We put 9:20 am and in the system and set 6:40 am as the user's input time.

h	passenger_count
0	1312
1	675
2	373
3	247
4	212
5	294
6	995
7	2496
8	4275
9	4795
10	5325
11	5731
12	6257
13	5974
14	6040
15	6146
16	5396
17	5616
18	5754
19	4772
20	3666
21	3502
22	2710
23	2094

Ex1)

- Rejection
- Input time: 9:20 am

```
At the chosen time, the current population is
4795

The maximum allowed safe capacity is
3527

Sorry user, you cannot make a reservation at:  9:20
Available time is:  [21, 22, 23]
The earliest time is:  21
```

Since the population at 9:20 am exceeds the threshold, the system rejected the reservation, and gave the user a list of available time and the earliest time.

Ex2)

- Acceptation
- Input time: 6:40 am

```
At the chosen time, the current population is
995

The maximum allowed safe capacity is
3527

Hurray! you are safe!
We made a reservation at: 6:40
```

Since the population at 6:40 am did not exceed the threshold, the system accepts the reservation.

Conclusion:

- There is a need for a system to help users book slots in buildings and public areas in order to enforce the etiquettes of social distancing during the COVID 19 pandemic. Therefore, we have designed a system to aid users in booking slots while ensuring that the distribution of slots is fair and based off of a curve generated by the data available to us.
- We preprocess the data and generate a demand model before the intervention of our proposed decision making system. Since we have taken a subset of the data to simplify the training period as an area in Central Park, Manhattan during a time period of September 2019, the model can be further extended easily by appending additional data to our dataset.
- Our optimal decision policy uses the model generated in (1) to decide whether or not to allow the user to book their slot. Our system has the flexibility of assuming a flat threshold for the capacity of the building, or it assumes the maximum allowed capacity is the average calculated over the curve of the graph.
- The system effectively ensures that the **graph has been flattened** by denying the user a slot if the capacity is over the threshold and searches the model for a time slot in the

future which would be safe for booking in increments of 1 hour. After suggesting the next safest time slot, the user can go ahead with the booking if they would like,

- Depending on the location ID code, a map is generated showcasing the available dropoff points in the vicinity.
- In this way, we have designed a system to not only use pre existing data available to the public, but have streamlined the process and made it into a user friendly system that successfully flattens the demand curve by ensuring people are not allowed into buildings above maximum capacity.