**IEOR E4404 Project Report - Spring 2021**

Group 9

Jiyin Chen (jc5498) Shuncheng Liu (sl4822) Gaole Lyu (gl2704) Fangzi You (fy2262)

**Project background:**

Since the start of year 2020, COVID-19 has wreaked havoc on the globe and lasted for a long time. Facing such pandemic, different countries came up with their solutions. As quarantine cannot be a long-term solution, vaccines are the only way out of this lasting pandemic. Hence, medical corporations have strived to research a few vaccines. In New York, Moderna, Pfizer, and Johnson & Johnson have launched their vaccines in many institutions and locations. As to see how efficient these vaccines are and the inoculation flow is to become, we bring up this project, in order to most closely study with the vaccines and perfect the solution for COVID-19.

**Value proposition of this project:**

As the COVID-19 has spread widely, vaccination is the only effective way to slow the spread rate and control the pandemic. In such a situation, increasing the efficiency of the vaccination process seems to be really important. For residents in New York, efficiency and rate of the vaccination will influence their safety level to great extent. Hence, in order to better boost the effectiveness, it is important to distribute the vaccination location, perfecting the inoculation flow. To the whole society, it is absolutely meaningful to perfect this system and ease the negative influence of this pandemic.

**The detailed description of the model:**

1. Overall system description & main function:
2. We create a multi-server-in-parallel and multi-environment system to simulate the arrival and vaccination processes for the five different boroughs in NYC, which are Brooklyn, Bronx, Queens, Manhattan and Staten Island. We model such a system by constructing our main function, **system\_sim**, which can be implemented in such a way:

In our model, we assume each borough as one independent environment, and people taking vaccinations in different boroughs form independent queuing processes, each having its own parameters (i.e., **lambda\_max** to be described later) to be passed as arguments for the **system\_sim** function. Also, the **capacity** set for each server represents how many nurses (at different vaccination sites all across one borough) work at the same time to help people get vaccinated. We have a given total number of vaccines for each borough, and use an **Output** dictionary to keep track of all arrival/vaccination times as well as the number of doses left for that borough. We then run the five parallel Arrival Processes using the five different environments already set up, until the five respective stopping times (all with same values equal to one day). In each Arrival Process, the Vaccination Process will take place one after another. Each time a person gets vaccinated and leaves, the number of vaccines left will decrease by one. The system output will give the sum of all left vaccines from these boroughs after simulating the system’s processes for one day.

1. After finishing system simulation, our model uses Tree-structured Parzen estimators (TPE) to find the minimum number of vaccines left by assigning different numbers of vaccines for each borough at the beginning of our process.
2. Arrival Process:

Based on data, we assume that the arrivals in the five boroughs have five different non-homogeneous Poisson processes, one for each. We find a lambda function for each borough to fit its respective data, and find out the **lambda\_max** based on this function so as to use the thinning method for simulation.The lambda function we find is assumed to have a linear relationship with time. We use those lambda values to generate people's arrival times for vaccination. Also, we create a random variable to decide people’s age (young people if less than 0.7, senior if otherwise) when they arrive, based on the data.

1. Vaccination Process:

Once people arrive, each will send a request to the server with different priorities based on their age. Old people have a higher priority than younger people. The number of vaccines left (which can be accessed with index [borough\_type]["vaccines\_left"]) will decrease by one. We get the stopping event **Stop\_sim** succeeded when no vaccine is left in this borough environment or when time has reached the total simulation time (one day), whichever comes first, thereby terminating the system right before any further request sent could ever be yielded and any more data could be added to the **Output** dictionary.

1. Optimization Process:

After multiple iterations, we find the best range for the number of vaccines distributed to each borough at the beginning are from 5,000 to 20,000 based on Appendices table 1. Thus, we created five parameters as the number of vaccines allocated to each borough at the beginning. The objective function is to minimize the total number of vaccines left. We use the **hyperopt.tpe.suggest** algorithm to evaluate 500 different sets.

**Result:**

*Analyze the system output:*

1. Output from analyzing the dataset:

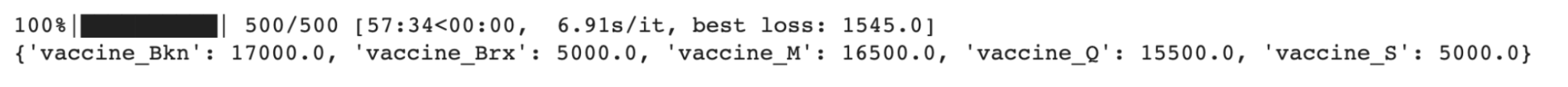
Since we are simulating the vaccine distribution for the incoming day and assume the arrival of people follows Non-Homogeneous Poisson processes for every borough, we need to find the relationship between vaccine number and time. By fitting a simple linear regression to the vaccine data, we can see each of the 5 boroughs have a downward trend in vaccination numbers since April 5. Therefore, we use the intercept of the linear regression model as our maximum lambda. And to interpret the coefficient of time, we think that the decreasing trend might be due to the coverage of the vaccine becoming larger or the overall infection situation is getting improved.

1. Output from simulation:

During the debugging progress, we printed the distribution of the number of vaccines left in each borough with an initial of 20,000 vaccines. From which we saw that almost every time that the stop event is triggered, the vaccine left in Queens or Manhattan will be the lowest, which is consistent with the real data. From Chart 2 we can see that the two boroughs, Manhattan and Queens, have the highest vaccination rate, which indicates that the vaccines given to these two boroughs should be more than those given to others.

*Perform policy experiment for policy recommendation and optimization:*

The system output is the sum of vaccines left from each borough with the given initial vaccine number. By applying the TPE optimization algorithm, we can find the optimal input value, which is the number of vaccines we need to distribute in each borough.



Based on the result, we can see that even though the overall vaccination rate of Brooklyn is low we still need to give them enough supply to satisfy a large number of people who want to get the vaccination. And for Manhattan and Queens, they still have high demand, which will take a large portion of the vaccine. However, for Bronx and Staten Island, the number of vaccines only need to meet their basic requirements.

**Discussion and conclusion:**

Contribution:

From the result of our simulation, we can help distribute the vaccine to achieve the optimal effects, perfect the inoculation flow, and ease the pandemic to the greatest extent.

Limitation:

As we can see in Chart 1, the vaccine number before April 5 should have an increasing trend. Therefore, we cannot use simple linear regressions to indicate the relationship between the number of vaccines and time. Unfortunately, we can’t find the vaccine data for each borough before that date to optimize the model and to make the arrival processes fit the real scenario.

**List the role and contribution:**

We divide the coding part, especially data preprocessing and system construction, and the final write-up part evenly for each group member. And the following contributions are for the tasks we listed in the proposal.

Jiyin Chen (jc5498): Build the system’s main structure, Improve the system’s logic

Shuncheng Liu (sl4822): Data collection, Improve the system’s logic

Gaole Lyu (gl2704): Build the system’s main structure, Improve the system’s logic

Fangzi You (fy2262): Data collection, Improve the system’s logic

**Links:**

*Final deliverable:*

<https://drive.google.com/drive/folders/1k5_BGZyyiEGPCO2KYZCiCBFvhs6dEcUk?usp=sharing>

*Dataset (we used):*

<https://drive.google.com/drive/folders/1PtLB2i8oa0yO8HxdD1fEmjifgeaNdW0y?usp=sharing>

*Link to online resources:*

1. NYC COVID-19:

<https://www1.nyc.gov/site/doh/covid/covid-19-data-vaccines.page>

1. Original COVID-19 dataset from github:

<https://github.com/nychealth/coronavirus-data>

1. Original vaccine dataset from github:

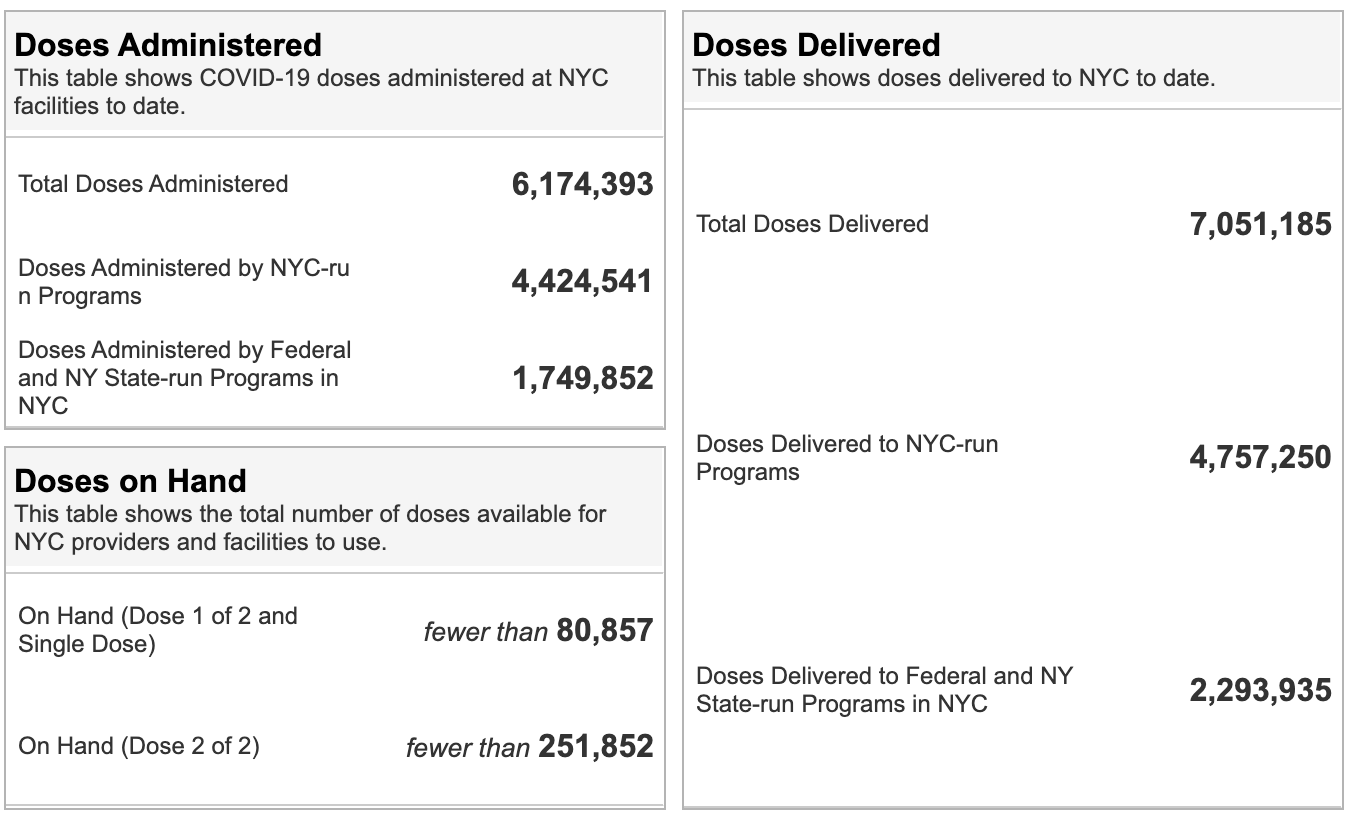
<https://github.com/bertrandmartel/covid19-nyc-vaccine-tracker/tree/master/deploy/2021-04-24>

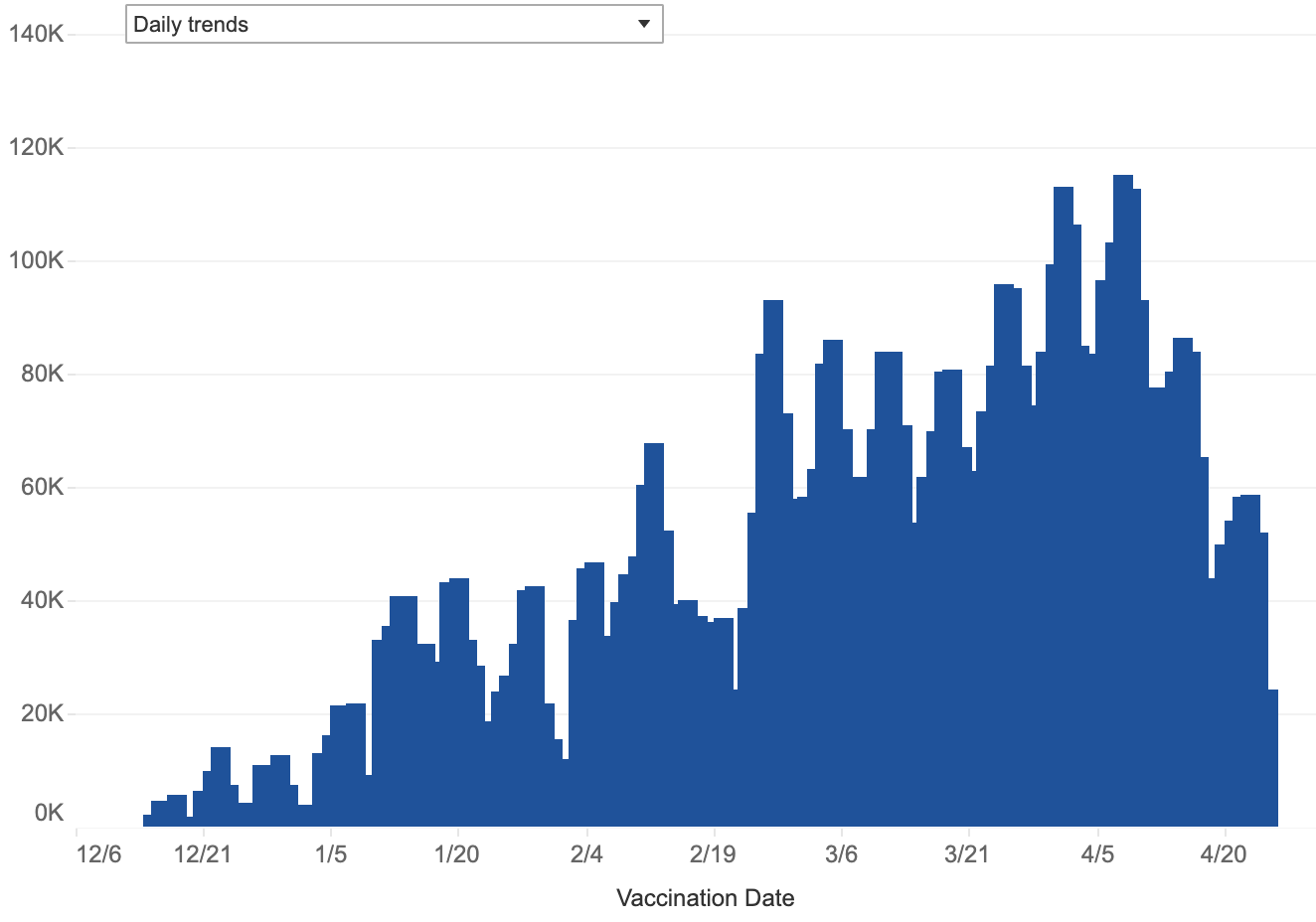
**Appendices:**

**Graphs and tables for analyzing:**

**From NYC Covid-19 website:**

**Table 1**



**Chart 1**

**Chart 2**

