Di Wang dw624
Yihao Luo yl3557
Yihe Chen yc2624
Yiqing Xin yx583
Yiwen Sun ys787
ORIE 5580 Final Project
12/10/2022

Designing an Volunteer Dispatch Policy to Increase the Survival Rate of Out-of-Hospital Cardiac Arrests

Executive Summary

When an Out-of-Hospital Cardiac Arrest (OHCA) happens, cardiopulmonary resuscitation (CPR) and defibrillation are two main factors influencing the survival rate of the patient, which are provided by ambulances. However, it takes up to 10 minutes for an ambulance to arrive at the scene of the emergency. In this case, volunteers who can provide CPR become a consideration to increase the survival rate of OHCAs. Thus, our purpose of this project is to develop a simulation analysis of volunteer dispatch to help determine how to design a dispatch policy in order to increase the survival rate of OHCAs. Theoretically, we should alert all surrounding volunteers, with the first volunteer to arrive at the scene delivering care, to increase survivability. However, if too many volunteers are notified, there will be too many volunteers showing up at the spot, which lowers the possibility for volunteers to respond to an alert in the long term. Therefore, we are asked to design dispatch policies to alert volunteers in a way that increases the likelihood of a response while avoiding alerting an excessive number of volunteers. The dispatch policies generated by our simulation should be efficient in that no strategy has a greater survival rate while maintaining alerted volunteers at the same level or lower.

To obtain the most suitable dispatch policy, we involve simulations to generate survival rates under different dispatch policies. According to the given information, the survival rate depends on two time-measures: the interval between the OHCA and the arrival of the first volunteer and the interval between the OHCA and the arrival of the ambulance. To simulate the time from OHCA occurrence to the show up of volunteers, given the location of a patient, the

simulation will only generate the locations of volunteers within a walkable distance, specifically, 1 kilometer. Then, by applying a dispatch policy, the simulation system will select a certain number of the nearest volunteers to send OHCA notifications. For each notified volunteer, the simulation will generate his responding time and use the calculated accepting rate to determine if he will come. The minimum arriving time among all arriving times of incoming volunteers will be the volunteer arrival time since the first arrival volunteer will provide CPR. Then, given an OHCA time, the simulation generates ambulance arrival time with a one-week Poisson distribution based on historical data of time from ambulance dispatch to arrival. Last, we calculate the average survival rate for a designed policy by repeating the process.

Then, by implementing a sensitive analysis that raises the number of notified volunteers from zero to a certain number, we could achieve a balance between the number of notified volunteers and the survival rate. Before the number of notified volunteers reaches 10, with one unit of notified volunteers increasing, the survival rate increases accordingly. And after then, the survival rate changes little, no matter how many volunteers increase. In fact, the most efficient dispatch policy is to send notifications to the nearest 8 volunteers surrounding an OHCA event, leading to a survival rate of 11.5%. To verify the dispatch policy and this low survival rate, we considered two extreme conditions. One is the worst condition when there is no volunteer arriving at the scene and giving support, which has a survival rate of only 7%. The other is the best condition when the volunteer stands next to the patient and provides CPR as soon as he receives the notification. Even in the best condition, it only has a survival rate of 21%. Furthermore, comparing the survival rate of the worst case with the proposed dispatch policy, we observed the support offered by volunteers only enhances 4.5% of the survival rate by. Therefore, the survival rate of OHCAs is always at a low level and does not largely depend on the help of volunteers. Developing a volunteer dispatch policy and using the notification app will not be efficient in improving the survival rate of OHCAs.

Problem Description

Out-of-Hospital Cardiac Arrest (OHCA) is a serious medical condition that involves the heart entering an atypical rhythm. It involves blood flow being cut off from the heart. In the absence of a medical response, the patient survival rate will fall rapidly. To greatly increase survival rate, high-quality cardiopulmonary resuscitation (CPR) and defibrillation (provided by an automated external defibrillator or AED) must be initiated within a few minutes, but the ambulance response time can take up to 10 minutes in cities, so increasing ambulance resources will not improve survival rate effectively.

This project aims to conduct a simulation analysis of volunteer dispatch to determine how to design the dispatch policy in a city in order to increase survival rates for OHCAs. Optimally, the policy creates a balance between the likelihood of response and the number of volunteers alerted to ensure the reliability of volunteer resources. In this way, the system decreases the probability of receiving passive, even little, responses derived from volunteers who arrived on scenes with nothing to do due to a large number of notified volunteers.

To be specific, one approach to develop a volunteer dispatch system is to notify volunteers only if they are within one kilometer of an OHCA. Upon receiving notifications, volunteers will make decisions on whether they will respond and travel to the scene to give CPR to patients till the ambulance arrives and provides defibrillation. However, instead of notifying all nearby volunteers, the system needs to make a careful selection and only alert some volunteers in a manner such that a response is very likely.

Modeling Approach and Assumptions

Before generating simulations, there are eight vital assumptions that need to be considered in our model. The first one is that there is only one OHCA at each replication of the model. The second is that volunteers can only provide CPR while ambulance crews can provide both CPR and defibrillation. The third assumption is a function of the survival probability of a patient. In the expression below, t_1 is the time in minutes until CPR is initiated, which equals the delay from alert to volunteer response plus the arrival time of the first volunteer. t_2 is the time in minutes until defibrillation is initiated, which is the dispatch time plus the arrival time of the ambulance.

$$(1 + e^{(0.04+0.3*t_1+0.14*(t_2-t_1)})^{-1}$$

The fourth is the distribution of time from cardiac arrest until an ambulance is dispatched and the volunteer software is initiated is a triangular distribution. Also, when a volunteer is notified, he may not respond. If he responds, he will only answer yes or no. Under such circumstances, the delay from alert to volunteer response can be estimated by a gamma distribution. The sixth assumption is that the fractions of volunteers that do not respond, accept, and decline after the alert are constant no matter the time of day or any other factors. Moreover, we assume 15% of the alerted volunteers that do not respond will come to the OHCA scene. Then the last assumption is that the rate of volunteers walking to the scene of an OHCA from their location is 6 km/h, and the walking path is a straight line.

With clear model assumptions, we develop the model with the following logic. First, we created patients' locations based on the distribution of OHCAs provided in the historical data of a similar city. Then, we divided the whole city map into 10x10 grids so that each grid has a length of 2.1 km. Assuming there are 10,000 volunteers in total, we generate the location of volunteers in the same grid and adjacent (at most 8) grids according to the distribution of citizens, which is the same as the distribution of OHCA, assuming the occurrence of OHCAs is independent of location. Next, we select volunteers in 1 km and record their distances to the patient and sort them in ascending order so that we get the first arriving volunteer. Then, we can generate a list of "True" and "False", which represents acceptance or rejection of each nearby volunteer. The acceptance rate is based on historical data, which is the addition of the accepted rate and 15% of no responding since there is a proportion of volunteers coming to the scene without responding to the alert system. Then, we element-wise multiply the list of distance and the list of TF based on the following rule. If we get a True, the corresponding distance is the true distance. If we get a False, we set the distance to a large number (e.g. 100000) so that if the volunteer doesn't come, we set the distance between this volunteer and the patient as infinity so that we will not consider this volunteer since we only choose the minimum time of arrival.

We first set k as 1 as our starting policy, which means we only alter one nearby volunteer. The alerting order is based on the distance between the patient and each volunteer. If he/she comes, we record his/her walking time. If not, the walking time will go to infinity. Then, we generate alert time and response time. By adding alert time, response time, and walking time together, we get a total value of arrival time and set it as the final volunteer arrival time. Next, we increase k to 2 and repeat the previous step for the second volunteer. If the second nearest volunteer's arrival time is smaller than the current time, update the final volunteer's arrival time as the second volunteer's arrival time. We record volunteer arrival time for k = 2. To test with different policies, we repeat the step until k is 50. Now we have volunteer arrival time under policies from k =1 to k = 50 for all the patients.

After getting the volunteer arrival time, we generate the ambulance arrival time for each patient based on historical data. Then, we calculate each patient's survival rate based on the given formula using two arrival times. Then, we generate a uniform random number between 0 and 1. If it is smaller than the survival rate, we record the patient as survival. If not, we record the patient as dead. Then, we take the average survival patient rate for each k. Finally, we find a certain k where the survival rate and the number of alerted volunteers reach a balance.

Data Analysis

Based on our modeling approach and assumptions, we need t1 and t2 to calculate the survival rate of patients. t1 depends on the location of the volunteer who responds yes and is closest to the OHCA patient. And t2 will be influenced by the time when an ambulance arrives and provides defibrillation. Therefore, to obtain t1 and t2, we need to find the following variables: alert time (time interval from cardiac attack until an ambulance is dispatched and the volunteer software is initiated), volunteer walking time (time interval between a volunteer responding yes and arriving to provide CPR), the volunteer response time (time interval between volunteer software initiation until a volunteer respond), and ambulance traveling time (time interval between ambulance dispatch and arrival).

Our estimation of these variables is based on two datasets. The first dataset contains 19,176 OHCA occurrences with locations in longitude and latitude and the corresponding ambulance dispatch time and arrival time. The second dataset records 11,301 data points of volunteer response time for each OHCA case.

To model the alert time, which is the time from cardiac arrest until an ambulance is dispatched and the volunteer software is initiated by the 911 ambulance officer, we chose a triangular distribution with a minimum, most likely value, and a maximum of 1.5 minutes, 2.5 minutes and 3.5 minutes based on common knowledge.

We used two steps to generate patient location. First, we grid cities into 10 * 10 squares of the same size and estimated the probability that a call falls into each square by calculating the proportions of OHCA occurrences in squares based on the dataset (Figure A). Based on these probabilities, we could generate a cumulative density function that we will use to determine in which square an OHCA will occur. Next, we assume that the calls are uniformly distributed within each square, which means that the calls will occur with equal probability at any point in a square. Therefore, once we determine the location of the square, we will choose a location uniformly from that square as the patient location.

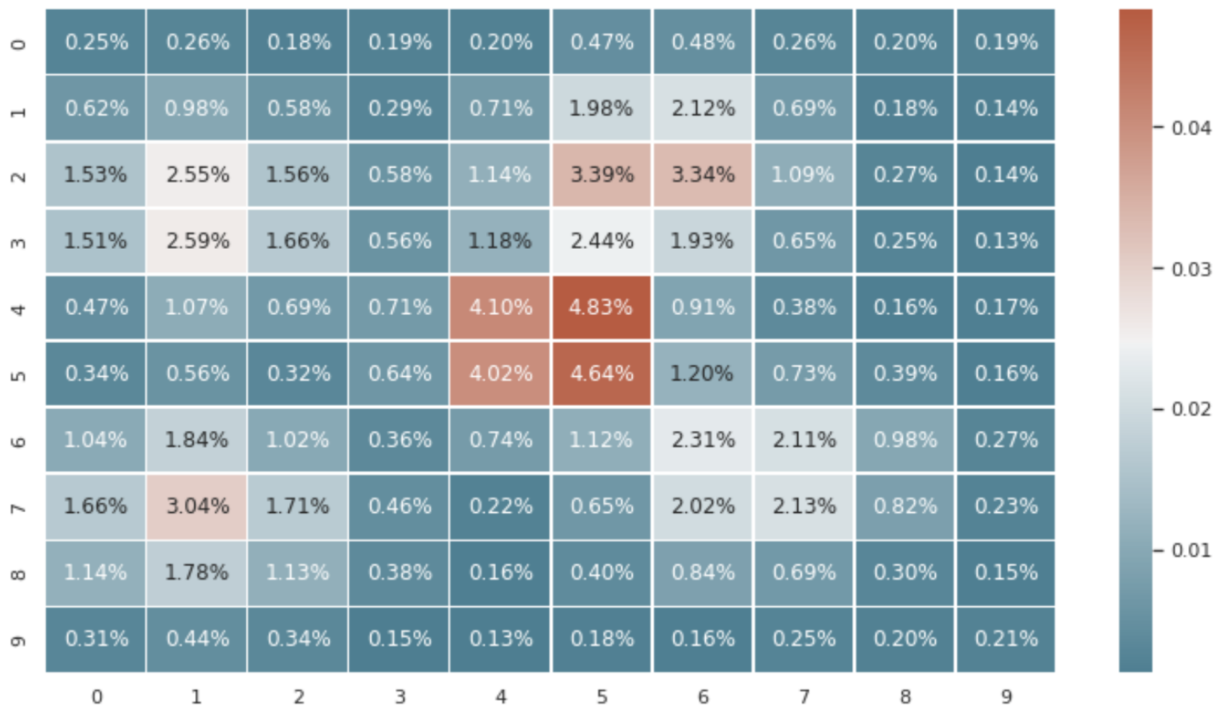


Figure A. Probability Map of OHCA Occurrence in City

We assume that the occurrence of OHCA is independent of location and time, which means the distribution of OHCAs follows the distribution of citizens in the city. Since we are not provided with the distribution of volunteers, we assume that the distribution of volunteers follows the distribution of citizens. Thus, both distributions correlate with the city's citizens' distribution. Generally, we will generate the distribution of volunteer locations based on the distribution of OHCAs since they all depend on citizen distribution. For each patient, volunteer locations are generated in all of the cells around the cell that the patient falls in. We generated volunteers within approximately 1 kilometer of the patient location.

Once we find all volunteers within 1 kilometer of the patient, we still need to determine if a volunteer will attend and make his or her way to the OHCA. The volunteer acceptance rate is calculated by the acceptance rate of the given dataset plus 15% of those who did not respond.

The last two variables we need to estimate are volunteer response time and ambulance traveling time. By analyzing the two datasets, we discovered that volunteer response time follows a gamma distribution, and the ambulance traveling time follows a lognormal distribution (Appendix Part I. d, e). However, due to large-scale datasets and independent locations and time for OHCA occurrence, we chose to estimate the response time for each volunteer and ambulance traveling time for each OHCA case by drawing responding data randomly from the datasets.

With the estimation methods of these variables defined, we could run a simulation analysis and determine the minimum number of volunteers to alert while keeping the survival rate as high as possible. Figure B shows survival rates with a change in the number of volunteers who are alerted, and the exact numbers with a 95% confidence interval are provided in Figure C. Orange line is the simulated survival rate, and the blue line demonstrates the true survival rates calculated from datasets. With the help of volunteers, the survival rate of OHCAs can reach a maximum of around 12%. With more volunteers alerted, the simulated survival rate increases drastically at first, then it gradually flattens. This finding suggests that when alerted volunteers reach a certain number, the survival rate will not have a significant increase even with a lot more volunteers informed.

In our simulated result, the survival rate flattens out when the number of alerted volunteers is around 7 to 9 and oscillates between 0.112 and 0.12. The highest simulated survival rate with 7 to 9 volunteers is around 11.7%. Therefore, to avoid alerting excessive volunteers, we will set the dispatch policy to alert the 8 closest volunteers around an OHCA patient, which ensures that a high survival rate can be achieved while keeping alerted volunteers at a reasonable level.

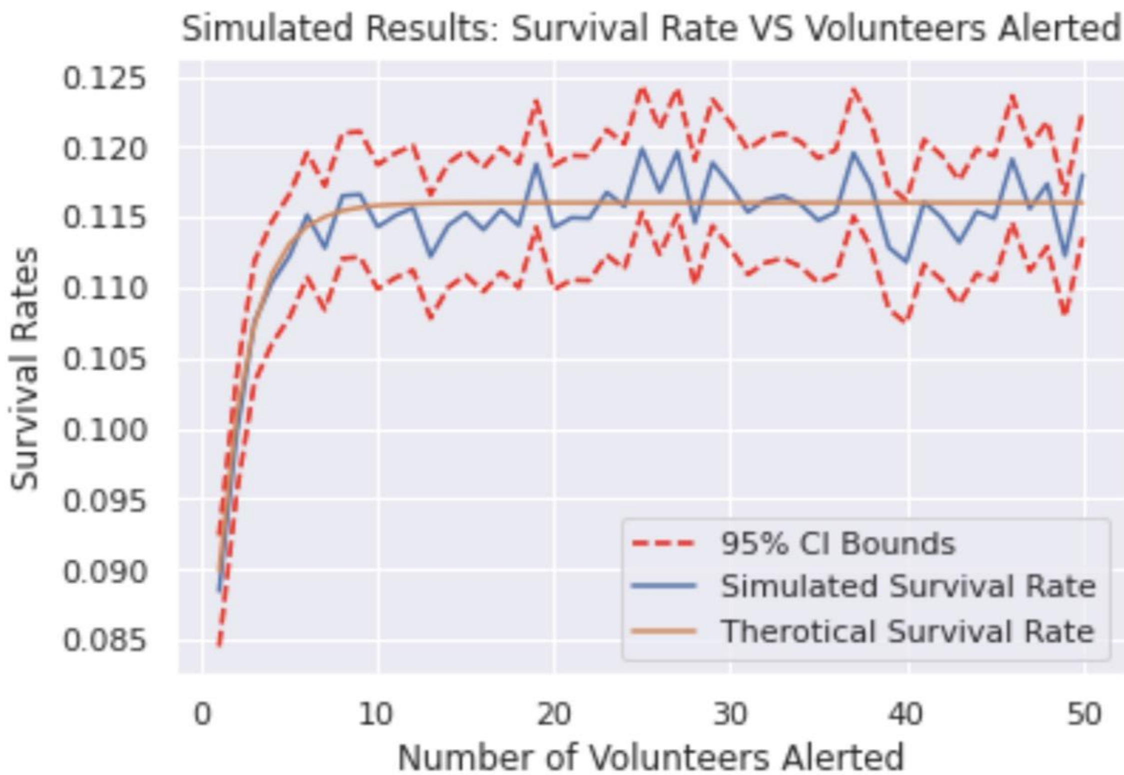


Figure B. The Line Graph of Survival Rate VS Number of Volunteers Alerted

	Volunteer Alteted	Theortical Result	Simulated Result	Simulated Result 95% CI LB	Simulated Result 95% CI UB
0	1	0.089848	0.09040	0.086426	0.094374
1	2	0.100895	0.10015	0.095989	0.104311
2	3	0.107218	0.11110	0.106745	0.115455
3	4	0.110833	0.11035	0.106008	0.114692
4	5	0.112957	0.11320	0.108809	0.117591
5	6	0.114215	0.11330	0.108907	0.117693
6	7	0.114861	0.11615	0.111709	0.120591
7	8	0.115299	0.11765	0.113185	0.122115
8	9	0.115523	0.11615	0.111709	0.120591
9	10	0.115662	0.11575	0.111316	0.120184
10	11	0.115736	0.11715	0.112693	0.121607
11	12	0.115776	0.11515	0.110726	0.119574
12	13	0.115818	0.11430	0.109890	0.118710
13	14	0.115840	0.11385	0.109448	0.118252
14	15	0.115851	0.11590	0.111464	0.120336

Figure C. The Table of Survival Rate With 95% Confidence Interval

Last, to find the most efficient number of notified volunteers when an OHCA happens, we implemented a sensitivity analysis that tests changes in survival rate based on unit increases in the number of volunteers. Specifically, we observed changes in survival rate from 0 notified volunteers to 50. Based on figure A, the survival rate oscillating around 11.5% has no big change after the number of volunteers reaches 8. To elaborate, the x-axis represents the number of volunteers while the y-axis represents the survival rate of an OHCA. The yellow line presents theoretical changes in survival rate while the blue line is survival rates generated by simulations. Both lines show similar results of survival rate. Figure B shows the specific number of volunteers and the exact digits of the survival rate. Clearly, after the number of volunteers reaches 8, the theoretical survival rate has a few increases and is stable at around 11.5%. At the same time, the simulated survival rate begins to oscillate around 0.115. Therefore, our volunteer dispatch policy suggests notifying the 8 nearest volunteers when an OHCA happens.

We proposed another policy that when the first 8 volunteers did not respond for more than two minutes, the nearest people other than the first volunteers were automatically notified. However, we found that after notifying the first eight people, with the likelihood of each individual not showing up being 0.6, the probability of eight people not coming at the same time is 0.6^8, or roughly 1.6%, which is extremely low. Additionally, the second group is informed two minutes after the first group, thus their arrival time would be two minutes later than the first group. Since the second group of volunteers is located further away from the patient than the first, there is little difference between their arrival time and that of the ambulance. As a result, the survival rate has not improved much with this policy.

Model Verification

To prove that our proposed volunteer dispatch policy will generate the best choice to balance the number of notified volunteers and the survival rate of OHCAs, we considered two extreme conditions. The first condition is when the arrival time of the first arriving volunteer is 0, which means the distance between the first arriving volunteer and the patient is zero kilometers. The second condition is when there are no volunteers, which means the survival rate only depends on the ambulance dispatch time. Theoretically, the survival rate of our proposed policy should be in the middle of these two extreme conditions.

To calculate the survival rate of the first condition, we assume the first arriving volunteer will provide CPR to the patient immediately once he receives the notification. The reason we still count the notification time is that we assume volunteers cannot give accurate support if they are not sure whether the patient got an OHCA or other diseases. By applying the equation below, with t1 equaling 0 and t2 equaling time from OHCA happening until the ambulance arrives, the final survival rate is around 21%.

$$(1 + e^{(0.04+0.14*t_2)})^{-1}$$

The other condition assumes the arrival time of volunteers is greater than the ambulance arrival time so that there is no volunteer to give the patient support. Without the help of volunteers, the survival rate of the patient will only be around 7%. Therefore, it is reasonable that our dispatch policy has a survival rate of 11.5%, which is between the two extremes. Moreover, since the worst condition still holds a 7% of survival rate, the help of volunteers seems to have little impact on the survival rate of OHCAs.

Conclusion

In designing the volunteer dispatch policy, we modeled a range of variables and ran 20,000 simulations with 40 million data points in total under a series of assumptions to find the optimal number of volunteers to alert. The result suggests that it is best to alert 8 closest volunteers around the patient in order to create a balance between the likelihood of response and the number of volunteers alerted through sensitivity analysis. The result is verified by looking at extreme scenarios at two ends of the survival rate spectrum. The volunteer dispatch system with our dispatch policy increases the patient survival rate from 7% to 11.5%. Although the number of patients who survive an out-of-hospital cardiac arrest has increased by 64% compared to an ambulance-only strategy, in practice this only results in a 4.5% rise. Thus, the influence of volunteers on the survival rate of OHCAs is positive but small.

Appendix

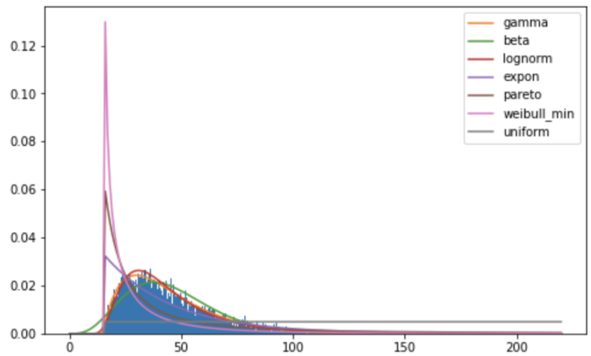
Part I. Input Generation

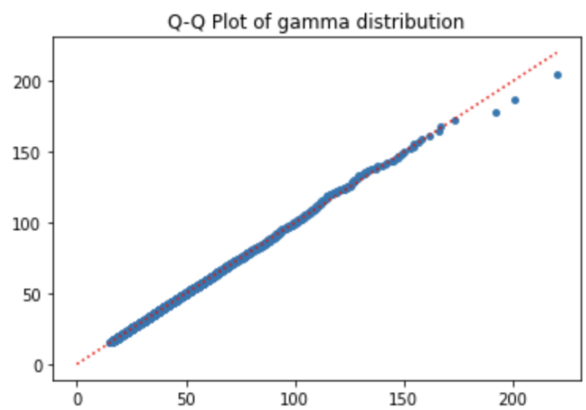
a) Activation volunteer generation method:
   - Assume total 10,000 volunteers in a city
   - We divide the city into a 10x10 grid with 1/10^2 probability that volunteers will occur in each grid. In this way, we generate volunteers randomly within 1km of an OHCA, where the OHCA is derived from the historical data.
   - Specifically, if the OHCA is in one grid, then the system will generate volunteers in all 9 nearby grids including 8 surrounding grids and one itself.
   - We limit the generated volunteers with historical average rate from volunteers who responded yes. And generated volunteers below this rate will not be selected as an activated volunteer.
   - Considering 15% of no response volunteers will come to the scenes as well, we add this part to the number of activated volunteers. Thus, the predicted activated volunteers include volunteers who will respond yes and 15% of volunteers who do not respond (The final probability of activated volunteers is 0.3915).

b) Generate period between OHCA happening and sending notification:
   - Triangular (1.5, 2.5, 3.5)

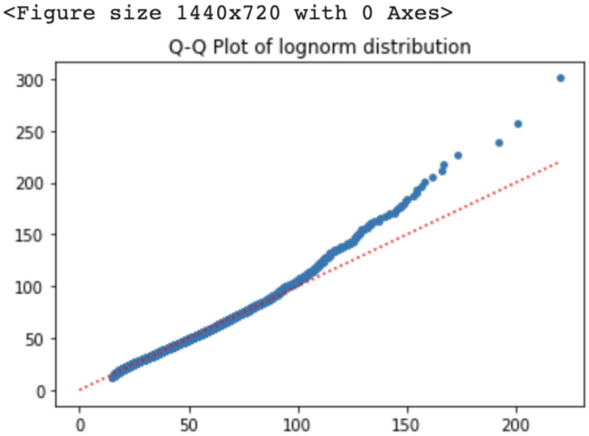
c) Generate period between sending notification and volunteers arriving:
   - Use response delayed data from a volunteer response database.
   - Randomly select delayed response time from volunteers who responded yes. The number of selections is equal to the number of generated activated volunteers.

d) To model the volunteer response delay, which is the time from alert to response over those volunteer alerts that received a response, We tested 6 different distributions on the modified data, including Beta, Gamma, Lognormal, Exponential, Pareto, and Weibull. Upon comparison, we found that Gamma and lognormal distribution fits the data the closest. To further distinguish between the two, we plotted qq plots for each distribution, and we conclude that Gamma distribution gives the best fit. The reason is that this distribution arises naturally in which the waiting time between Poisson distributed events are relevant to each other.

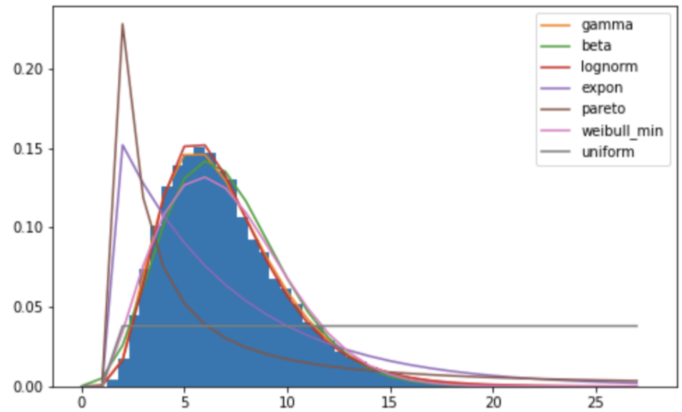




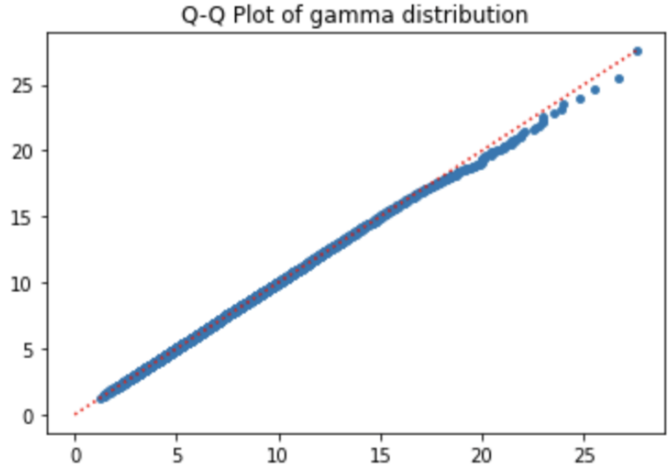
<Figure size 1440x720 with 0 Axes>



e) Ambulance traveling time (ambulance arrival time - ambulance dispatch time): random draw. To model the time for ambulance arrival, which is the time difference between ambulance arrival time and ambulance dispatch time, we used the same fitted package to test 6 different distributions on the modified data as we did to model the delay from alert to volunteer response. Based on fitted outputs, gamma and lognormal distributions seem to fit data the best. To identify which one is better, we plotted Q-Q plots and found that lognormal distribution gives the best fit.

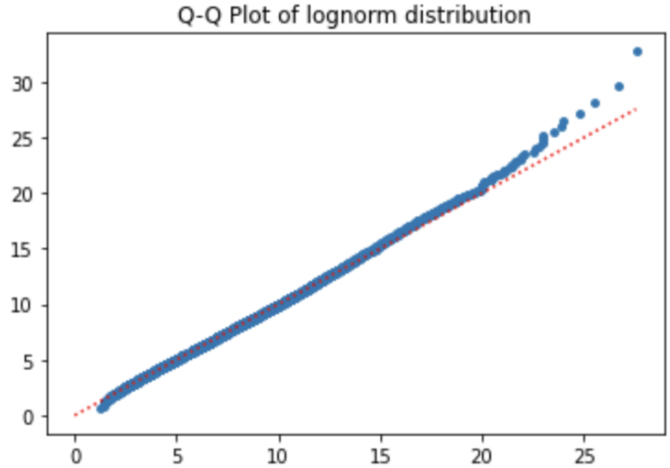


Q-Q Plot of gamma distribution



<Figure size 1440x720 with 0 Axes>

Q-Q Plot of lognorm distribution



Part II. Important Code Function Definitions:

a) Gen_Nearby_Volunteers (Num_Available, Activation_Rate, Map, Patien_Relative_Location):
   - This function will generate the number and locations of volunteers in a city who will go to the OHCA scene.
   - Num_Available: the number of available volunteers in a city
   - Activation_Rate: the rate of volunteers who receive the OHCA notification going to the scene. In our model, this includes both volunteers responding yes and 15% of volunteers not responding.
   - Map: also called area density, is an n by m grid of the city map with one unit length of x and y. This is related to the other function: create_matrix(row, col, x_Lon, y_Lat), which generates the map of a city.
     - create_matrix(row, col, x_Lon, y_Lat)
     - Row: The number of rows of the generated map grid
     - Col: The number of columns of the generated map grid
     - x_Lon: a list of all unit longitude of a city decided by min and max of longitude
     - y_Lat: a list of all unit latitude of a city decided by min and max of latitude
   - Patient_Relative_Location: (x, y), where x,y in [0,1], is the patient grid location inside the n by m grid of the city map

b) gen_ambulance_travelling_time ():
   - Randomly draw an ambulance traveling time from dataset

c) gen_volunteer_response_time (accept = 0):
   - If volunteer respond yes:
     - Randomly draw response time from all volunteers who respond yes
   - If volunteer respond no:

  - Randomly draw response time from all volunteers who respond yes
- If volunteer did not respond:
  - Randomly draw response time from all volunteers who did not respond

d) gen_volunteer_arrival (Num_Available, patient_location):

#generate number of volunteers within 1 km and the minimum traveling time for a patient

- Num_Available: the number of available volunteers in a city
- Patient_Location: locations of patients
- Invoke Gen_Nearby_Volunteers function to get the list of patient and volunteer locations
- Set response time to the responding time of volunteers who respond yes.
- For each patient:
  - Distance = distance from all volunteers to this patient (in km)
  - If distance <= 1 km:
    - Num_Within_1km += 1
    - Volunteer_travelling_time = distance / 6 * 60
    - Min_Arrival_Time = Min(volunteer_travelling_time)
- Return Num_Within_1km, Min_Arrival_Time

e) gen_weekly_data_all_activated (Num_Volunteers):

#generate weekly_data table of OHCA occurrences with variables needed and t1, t2

- Generate one week of patient locations
- For each patient, create dictionary keys and value entries for Dispatch_Time, Num_Activited, Volunteer_Arrival_Time, Ambulance_Travel_Time, t1, and t2
  - Dispatch_Time = np.random.triangular (1.5, 2.5, 3.5)
  - t1 = Volunteer_Arrival_Time + Dispatch_Time
  - t2 = Ambulance_Travel_Time + Dispatch_Time
- Return weekly_data

f) lowest_among_first_n_terms(input_list):

- Input_list: a list of volunteers alerted time

- This function returns the rank of alerted time of first alerted volunteers in each OHCA event.

g) trim(input_list, K):
  - Input_list: a list of volunteers alerted time
  - K: the number of volunteers alerted
  - The function is used to avoid generating data errors since the number of generated volunteers surrounding one patient is different.
  - The function returns the same size of generated volunteers from different OHCAs.

h) first_arr(num_activated, N):
  - N: simulation times, which is 20,000 in our case
  - num_activated: the number of volunteers who go to the OHCA scene
  - The function returns the arrival time of the first arriving volunteer based on the given alerted time

Part III. Brief Simulation Logics:

- For 20,000 (N = 20,000) OHCAs with patients i:
  - For 2,000 volunteers of each patient i:
    - gen_weekly_data_all_activated (Num_Volunteers) returns t1 for each first arriving volunteer and t2 for each arriving ambulance of each OHCA.
    - Calculate survival rates using t1 and t2