AoCMM 2016 - #822
The 2016 Annual Association of Computational and Mathematical Modeling Essay Sheet

Contents

AoCMM 2016 - #822

Problem 1

Best Financial Strategy for U.S. Presidential Campaign

Summary

During our investigation of finding out an optimal strategy to divide the financial resources of one of two candidates to each state to maximize the probability of winning the election, we basically built three mathematical models based on normal distribution in statistics, the law of diminishing marginal utility in economics, the plurality voting system ("winner takes all") in U.S. politics. By combining these three models, we eventually gave out the solution through algebraic methods.

The FIRST model of our team is about normal distribution. We assumed that the certainties of each individual voter in state X to a candidate are normal distributed on the probability density function of the support rate of this candidate in state X before candidates plunge their money into campaign. Known the certainties and the support rate before spending financial resources within state X, we can calculate the percentage of population of state X who holds a particular certainty (certainty β) to the candidate we represented. The SECOND model of our team about marginal utility shows the relation between effective financial resources, results of the historical percentage of elections' results that the party of each candidate won and two financial resources spend on campaign, and the change of individuals' certainties within a state from the particular certainty (certainty β) to the final certainty that group of voters hold when they go to the poll stations. We applied the concept of the law of diminishing marginal utility because we thought that as the money spent increases, the certainty increases, too, but the rate of increasing decreases. Thus we used exponential function to embody the diminishing marginal utility. The THIRD model of our team is about "winner takes all". For each individual voter, we believe that if his/her certainty to a candidate is larger than 50%, he/she will choose to vote for that candidate, vice versa. For each state, according to the plurality voting system, if the final support rate of a candidate of a state is up to 50%, all citizens of that state in the Electoral College will vote for that candidate. To realize these conditions, we applied segmented functions.

In addition, we conducted sensitivity analysis by comparing how different values of constants (k=0.986; $\varphi = 1.1$) we set affect the three models. By this operation, we found out the best values for these constants. The results of the calculation for these values of constants are so fitted to our hypotheses that it convinced us that our models are reliable and accurate.

With the restriction of the three models as an equation set, our goal eventually turned out to be to maximize the expression of the sum of the voters in the Electoral College from all fifty states. In other words, we altered this question into equations and expressions, and we applied algebraic method to solve them, thus finding out the best strategy to distribute the financial resources of a candidate.

I. Introduction and Problem Restatement

1.1 Introduction

Campaign finance plays a critical role in every U.S. presidential campaign. It affects candidates' ability to stay in the race and the possibility to win the election. As in the corporate world, no candidate can compete without continuing and sustained financial support. Money may not buy elections but without it, you can't even be in the running. (Kirby Goidel & Keith Gaddie, 2016) Money really matters in presidential elections!

A mathematical method that maximizes the possibility for one of the two major candidates to win the campaign is likely to stand out from manual estimations. In fact, there are too many uncertainties in a presidential campaign that even mathematicians cannot expect, but what we can do is to try our best to consider all the uncertainties and work out a general solution for candidates except outliers who do not conform with the standard of normal candidates.

We concerned mostly about financial resources as the question has requested. Financial resource is more about economics; consequently, we applied some basic ideas in economics to build our models. However, because the number of voters is huge, we actually used more statistical methods to solve this question.

U.S. presidential election is one of the most important political events in the world. We believe that our models and methods can effectively help a candidate establish a mechanism to allocate his/her financial resources to 50 states in the United States to maximize his/her possibility of winning the election and becoming the president.

1.2 Problem Restatement

Our condition is that we know the certainty to candidate A and support rate of candidate A in all fifty states in the United States of the polls from the very beginning of the election to the time when candidate A starts to plunge huge amounts of his financial resources into the campaign. We also know how many times each of the fifty states votes for the Democratic candidates or Republican candidates. Our goal is to find out the best way to distribute the financial resources of candidate A nationally for purpose of maximizing his probability of winning the presidential campaign with the restriction of the fixed total financial resources, the "winner takes all" system and electoral college system.

II. Definitions, Justifications, Assumptions and Variables

2.1 Definitions

Candidate A: the one we chose out of the two candidates that are running the presidential campaign.

Certainty (to candidate A): The percentage of an individual voter's extent to support to candidate A.

Support rate (to candidate A): The percentage out of 100% of a state level polls that shows how many people out of the whole population within that level of unit (especially a state level) support to candidate A.

Probability Density Function: a function that describes the relative likelihood for this random variable to take on a given value (Wikipedia: Probability Density Function).

Normal Distribution: a representative of real-valued random variables whose distributions are not known (Casella & Berger, 2001, p. 102).

Utility: a measure of preferences over some set of goods and services (Wikipedia: Utility)

Law of Diminishing Marginal Utility: the first unit of consumption of a good or service yields more utility than the second and subsequent units, with a continuing reduction for greater amounts (Wikipedia: Marginal Utility)

Winner takes all (Plurality voting system): a voting system in which each voter is allowed to vote for only one candidate, and the candidate who polls more votes than any other candidate (a plurality) is elected (Wikipedia: Plurality voting system)

Electoral College: the body that elects the President and Vice President of the United States every four years. Citizens of the United States do not directly elect the president or the vice president; instead they choose "electors", who usually pledge to vote for particular candidates. (Wikipedia: Electoral College (United States))

2.2 Assumptions and Justifications

In order to streamline our model, we have made several key assumptions.

Assumption: All citizens in a state voted for either of the two candidates.
Justification: We considered that no American citizen gives up his/her right to vote for a president.

Assumption: The cases applied to candidate A is identical as the one applied to Candidate B.
Justification: The question stated that our team should represent one of the two candidates, which means that we can choose either of the two persons because there is no difference.

Assumption: Candidate A, whom we considered in our case, knows the financial resources candidate B would plunge in all states, but candidate B does not know anything about how candidate A is going to allocate his/her resources among fifty states. Candidate B also does not know the total financial resources candidate A has.
Justification: Candidate A, whom we are going to help to win the election, has not decided his strategy of how to use his money yet. Even candidate A himself does not know how he is going to distribute his money, not to mention candidate B!

Assumption: There are only two candidates in the presidential election finally.
Justification: Although in reality, there are more than two candidates, the candidates other than the major two seldom obtain more than 5% of ballots. To simplify our model, we didn't consider the existence of these candidates.

Assumption: Two candidates basically follow the styles of their party.
Justification: Because we considered how much a state support a particular party, two candidates have always to follow the styles of their party so as to be consistent for our model.

Assumption: The certainties of all individuals in a state are normal distributed in the normal distribution of support rate of each candidate in this state.
Justification: Certainties of all individuals fit the definition of normal distribution because they are real-valued random variables, and the support rate of a state is the representative of their collective result.

Assumption: Each candidate's financial resources affect the citizens within a state as a collective unit.
Justification: With the development of technology, more and more information can be easily disseminated to everyone in every corner of the earth. Here we considered that when a candidate spent his/her money on a state, all the citizens in this state will be affected because of the modern technology.

Assumption: If an individual voter's certainty to candidate A is larger than 50%, this voter eventually will choose candidate A, vice versa.
Justification: This is another definition of certainty. "Larger than 50% to candidate A" means that the certainty to candidate A is bigger to that to candidate B because their sum is 1. For an individual voter, he/she only concerns about which candidate he/she

supports most. As a result, we only considered the conditions of larger and smaller than 50%.

Assumption: The financial resources plunged into TV advertisements are proportional to all the money two candidates spend within a state.
Justification: We can only get credible data of how many financial resources two candidates plunged into TV advertisements in 2012 U.S. presidential campaign; thus, by assume this, we can better compare cases of different states in 2012 campaign and get the value of specific constants in our models.

Assumption: We ignored the case that the percentage of the certainties to both candidates of an individual voter are both 50%.
Justification: This is an impossible case in reality because voters cannot choose both of the two candidates and they must make their choice.

2.3 Variables

σ_n is the standard deviation of the results of all the polls in state n before both candidates plunge their financial resources into the campaign.

β is a particular individual certainty we chose from the normal distribution of the result of polls of candidate A before both candidates plunge their financial resources into the campaign.

$f(\beta)_n$ is the function showing that how many percentages of the population of state n holds that certainty β.

μ_n is the support rate of candidate A in state n before both candidates plunge their financial resources into the campaign.

$T(\beta)_n$ is the percentage of the population of state n who holds the certainty of β to candidate A.

$K(z_n)$ shows how certainty of the people who at first holds certainty of β to candidate A changes as z changes.

z_n is the effective financial resources to the growth of individuals' certainties in state n.

$G(T(\beta)_n)$ represents the percentage of the total population of state n that in the group of the people who at first holds certainty of β to candidate A decides to vote for candidate A eventually.

a_n is the weighted support rates to candidate A from the perspective of history under the consideration of the parties behind candidate A; b_n is the weighted support rates to candidate B from the perspective of history under the consideration of the parties behind candidate B.

d_{an} represents the percentage of elections' results that the party of candidate A won from 1872 to 2012; d_{bn} represents the percentage of elections' results that the party of candidate B won from 1872 to 2012.

e_{an} represents the percentage of elections' results that the party of candidate A won from 1980 to 2012; e_{bn} represents the percentage of elections' results that the party of candidate B won from 1980 to 2012.

$a_n + b_n = 1$; $d_{an} + d_{bn} = 1$; $e_{an} + e_{bn} = 1$

φ is a constant around 1.1; k is a constant around 0.986.

$e(n)$ represents how many voters of state n are in the Electoral College.

$V(n)$ is the ballots candidate A receives from state n.

III. Mathematical Models

3.1 Categorize voters within a state by their different certainty to candidate A

We assumed that our team is candidate A, and we studied the case within a state first. Suppose the state is Ohio (the number that represents Ohio is n; the case study of Ohio is true to all the fifty states in the United States). Plus, the question stated that the candidate knows his support rate. Thus, we considered that both candidates know their support rates before plunging their financial resources, and for each candidate, he/she not only knows his/her support rate in the recent polls, but also in all polls from the very beginning of the election. By calculating these support rates of candidate A in our case, we can obtain the standard deviation of them. The formula of standard deviation is:

$$\sigma_n = \sqrt{\frac{\sum_{i=1}^{m}(\alpha_m - \overline{\alpha})^2}{m}}$$

σ_n is the standard deviation; m is the number of the samples (the number of polls); α_m is the support rate of the poll before candidate A and candidate B plunge their financial resources; $\overline{\alpha}$ is the average of the support rates of candidate A in all polls.

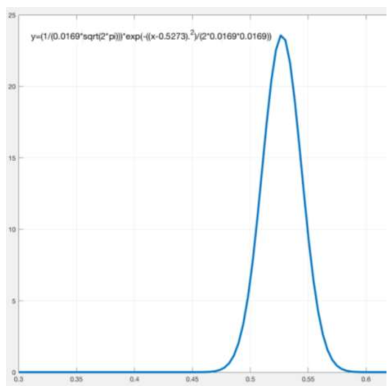
By working out the standard deviation for all polls and knowing the support rate of candidate A just before he/she plunges his/her money into the campaign, we can obtain the probability density curve on the normal distribution of the support rate of candidate A. We considered that the certainties of all individuals in a state are normal distributed in the normal distribution of support rate of candidate A in Ohio (actually in every state in the United States). Utilizing the probability density curve, we actually show that for any particular individual certainty (suppose it is β), the percentage of the population of a state holds that certainty (β).

For certainty β (to the nearest 0.01%), the value of the probability density curve is:

$$f(\beta)_n = \frac{1}{\sigma_n\sqrt{2\pi}} e^{\frac{-(\beta-\mu_n)^2}{2\sigma_n{}^2}}$$

β can be any possible certainty between 0 to 1; σ_n is the standard deviation; μ_n is the support rate of candidate A in Ohio, the state we studied here.

Figure 1: The normal distribution of all individual voters' certainties in Ohio in 2012 U.S. presidential campaign



Considering 0.1% as a unit of certainty, we applied definite integral to calculate the percentage of Ohio's population that holds the certainty to candidate A from $\beta - 0.05\%$ to $\beta + 0.05\%$. Subsequently, we regarded this percentage of Ohio's population as the number of people who holds certainty of β to candidate A, the formula of which is:

$$T(\beta)_n = \int_{\beta-0.05\%}^{\beta+0.05\%} \frac{1}{\sigma_n\sqrt{2\pi}} e^{\frac{-(\beta-\mu_n)^2}{2\sigma_n{}^2}} \, d\beta$$

σ_n is the standard deviation; μ_n is the support rate of candidate A in Ohio, the state we studied here; $T(\beta)_n$ is the percentage of Ohio's population who holds the certainty of β to candidate A.

3.2 The relation between financial resources that candidate plunges into a state and the increase of the certainty of a particular group of individual voter within a state to candidate A

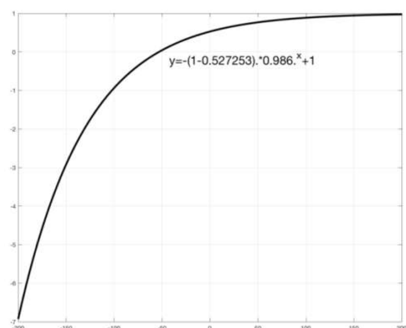
3.2.1 General model about the relation

In this section, besides restricting that our team is candidate A and that we only studied Ohio (any one state in the United States), we also stipulated that we only considered the group of voters who held certainty of β to candidate A. According to economics, the utility (effect) of financial resource of candidate A is by no means proportional to the rise of an individual citizen's certainty to candidate A. Rather, it follows the law of diminishing marginal utility: the first unit of the use of money to Ohio (any one state) in the campaign yields more increase in all individuals certainties within Ohio (any one state) than the second and subsequent units, with a continuing reduction for greater amounts. Therefore, referring to this economic law, we managed to build the relation between the financial resources candidate A plunges into Ohio and the certainty's increase of the group of voters who hold certainty of β to candidate A, which is:

$$K(z_n) = -(1-\beta)k^{z_n} + 1, k \in (0,1)$$

z_n, which will be defined by formulas in the following passage, is the effective financial resources to the growth of individuals' certainties in Ohio; β is the previous certainty; k is a constant, the value of which is around 0.986 (we will prove it in the following passage); $K(z_n)$ shows how certainty of the people who at first holds certainty of β to candidate A changes as z changes.

Figure 2: The relation between certainty and financial resource in 2012 U.S. presidential campaign in Ohio



X-axis represents the value of z_n or, in other words, the amount of effective financial resources (in millions of U.S. dollar); Y-axis represents the certainty.
Source: Washington Post, Huffpost pollster and http://www.270towin.com/

The function of the marginal utility of the use of financial resources is:

$$K'(z_n) = -(1-\beta)k^{z_n} \times \ln(k), k \in (0,1)$$

3.2.2 What does z actually represent?

z_n, according to the definition, is the effective financial resources to the growth of individuals' certainties in Ohio (any one state). However, what does "effective" exactly mean here? Here we set a model for effective financial resources, which is:

$$z_n = {a_n}^{\varphi} \times M_n - {b_n}^{\varphi} \times Q_n$$

$$a_n = \frac{d_{an}+e_{an}}{2}; \; b_n = \frac{d_{bn}+e_{bn}}{2}$$

d_{an} and d_{bn} represent the percentage of elections' results to the party of candidate A and that to the party of candidate B respectively from 1872 to 2012; e_{an} and e_{bn} represent the percentage of elections' results to the party of candidate A and that to the party of candidate B respectively from 1980 to 2012. By calculating their mean, we can obtain the weighted support rates to both of two candidates from the perspective of history under the consideration of the parties behind these two candidates. Respectively, there are a_n for candidate A's party and b_n for candidate B's party. $a_n + b_n = 1$. M_n represents the financial resources candidate A plunges into Ohio (any one state); Q_n represents the financial resources candidate B plunges into Ohio (any one state). φ, which is around 1.1, is tested by mathematical calculations, the detail of which will be explained in the following passage.

3.2.3 Why does "k" equal to 0.986?

From the expression $K(z_n) = -(1-\beta)k^{z_n} + 1, k \in (0,1)$, we can know that $k = \sqrt[z_n]{\frac{1-K(z_n)}{1-\beta}}$. By imputing data of 2012 presidential election, we can get the value of k. The details of how to get the value of k will be shown in the following passage.

3.3 The relation between a voter's certainty to candidate A and the election result within a state

We assumed that if an individual voter's certainty to candidate A is larger than 50%, this voter eventually will choose candidate A, vice versa. We also ignored the case of 50-50. By these two assumptions, we can know that as long as $K(z_n)$ is larger than 50%, the group of citizens in Ohio (any one state) who at first (before candidates plunge their money into campaign) holds the certainty of β to candidate A will vote for candidate A. The function of this relation is:

$$G(T(\beta)_n) = \begin{cases} T(\beta)_n, K'(z_n) \in [0.5,1) \\ 0, K'(z_n) \in (0,0.5) \end{cases}$$

$G(T(\beta)_n)$ represents the percentage of the total population of Ohio (any one state) that in the group of the people who at first holds certainty of β to candidate A decides to vote for candidate A eventually.

3.4 The relation between an election's result within a state and the national result

After obtaining the final decision of specific groups in Ohio, we can add the results from these groups together to judge that whether their sum is larger than 50%. If the sum is bigger than 50%, according to © 1998-2016 ACE project and Wikipedia: Plurality voting system, all the Ohio's people in the Electoral College will choose candidate A eventually, vice versa. The function of how many Ohio's voters in the Electoral College vote for candidate A is:

$$V(n) = \begin{cases} e(n), & \int_0^1 G(T(\beta)_n)\, d\beta \in (0, 0.5) \\ 0, & \int_0^1 G(T(\beta)_n)\, d\beta \in (0.5, 1) \end{cases}$$

e(n) represents how many Ohio's voters are in the Electoral College; $V(n)$ represents the how many Ohio's voters in the Electoral College eventually vote for candidate A. Actually, here $\int_0^1 G(T(\beta)_n)\,d\beta$ should be considered as $\sum_{\beta=0}^{1001} G(T(\beta)_n)$ with the interval of 0.1%. This segmented function basically shows that the presidential election in the United States follows the law of "winner takes all" within a state.

$V(n)$ is the ballots candidate A receives from Ohio; there are fifty states in the United States and we need to add them altogether to get the total ballots candidate A got in the Electoral College. The function of the total ballots candidate A can get can be expressed as:

$$W_A = \sum_{n=1}^{50} V(n)$$

W_A represents the total ballots in the Electoral College candidate A can get; when W_A is larger than 270 out of 538, candidate A can become the president, vice versa.

3.5 Expressions for finding out how to allocate the financial resources of a candidate out of two

So far, the question altered into a problem of maximizing $W_A = \sum_{n=1}^{50} V(n)$ known that:

$$\begin{cases} \sum_{n=1}^{50} M_n = t \\ z_n = a_n{}^{\varphi} \times M_n - b_n{}^{\varphi} \times Q_n \\ K(z_n) = -(1-\beta)k^{z_n} + 1, k \in (0,1) \\ T(\beta)_n = \int_{\beta-0.05\%}^{\beta+0.05\%} \frac{1}{\sigma_n\sqrt{2\pi}} e^{\frac{-(\beta-\mu_n)^2}{2\sigma_n{}^2}}\,d\beta \\ G(T(\beta)_n) = \begin{cases} T(\beta)_n, K'(z_n) \in [0.5,1) \\ 0, K'(z_n) \in (0,0.5) \end{cases} \\ V(n) = \begin{cases} e(n), & \int_0^1 G(T(\beta)_n)\,d\beta \in (0, 0.5) \\ 0, & \int_0^1 G(T(\beta)_n)\,d\beta \in (0.5, 1) \end{cases} \end{cases}$$

The variables we input into this equation set is "t" as the total financial resources candidate A possesses; "a_n" as the aforementioned weighted support rate to candidate A from the perspective of history under the consideration of the parties behind him/her; "b_n" as the aforementioned weighted support rate to candidate B from the perspective of history under the consideration of the parties behind him/her; "φ" as a constant around 1.1; "β" as the aforementioned certainty to candidate A we took out as an example; "μ_n" as the support rate of candidate A before both of the candidates plunge their money; "k" as a constant around 0.986; "$e(n)$" as the number of Ohio's voters who are in the Electoral College. Additionally, for the sake of simplicity, we treat "Q_n", the financial resources candidate B plunges into each state, as a known factor. By these input values, we can eventually find out the optimal way to allocate the financial resources among fifty states in the United States.

IV. Sensitivity Analysis

To determine the effectiveness of our mathematical models, we conducted our sensitivity analysis by adjusting the constants in the models, such as k and φ. However, first of all, we endeavored to show that how we set these two values with certain reasons.

In the mathematical models we built, we defined k=0.986 as a constant and φ=1.1 as a constant, too. However, different value of k and φ may affect the final result of our models. To test the sensitivity of these constants, we applied the data of 2012 U.S. presidential election. According to the data from Washington Post and Huffington Post, we created table 1 to show how much money Obama and Romney, two presidential candidates in 2012, spent on their campaign (e.g. in TV advertisements), their support rate in the ten states they plunged money into before and after their financial resources being used and the standard deviation of either of their support rates from all polls in the campaign.

Table 1: Campaign spending, support rates and standard deviation of 2012 U.S. presidential election

State	Obama's spending from April to Nov.	Romney's spending from April to Nov.	Obama's support rate in April	Romney's support rate in April	Obama's final result in Nov.	Romney's final result in Nov.	Standard Deviation of Obama/Romney
FL	78	95	0.513158	0.486842	0.504541	0.495459	0.018837146
VA	68	83	0.52381	0.47619	0.519797	0.480203	0.021438574
OH	72	78	0.527253	0.472747	0.515244	0.484756	0.01693041
NC	40	57	0.50385	0.49615	0.489879	0.510121	0.018936905
CO	36	37	0.52381	0.47619	0.527664	0.472336	0.016788607
IA	27	30	0.515084	0.484916	0.529532	0.470468	0.017191565
NV	26	29	0.531729	0.468271	0.534149	0.465851	0.013922747
WI	13	27	0.53787	0.46213	0.534954	0.465046	0.021883886
NH	18	16	0.536264	0.463736	0.528455	0.471545	0.028923602
MI	8	24	0.540601	0.459399	0.548028	0.451972	0.025238991

Source: Washington Post for financial resources; Huffington Post for support rates

Subsequently, we considered the final result of Obama in November to be the percentage of citizens within a state whose voting certainties to Obama are larger than 50%. In the expression $K(z_n) = -(1-\beta)k^{z_n} + 1$, we supposed $K(z_n)$ to be 50% and β to be Obama's support rate in April. Then $50\% = -(1-\beta)k^{z_n} + 1$ is the general equation while $50\% = -(1-0.513158)k^{z_n} + 1$ is the equation for the case of Florida. By these operations, we could know that $k = (2-2\beta)^{-\frac{1}{z_n}}$ while $z_n = {a_n}^{\varphi} \times M_n - {b_n}^{\varphi} \times Q_n$. β is a constant for each state, so the only thing we need to know before solving the value of k is z_n. Table 2 shows the value of a_n and b_n in the 10 states.

Table 2: The value of a_n (weighted average value of historical support rate to candidate Obama's party) and b_n (weighted average value of historical support rate to candidate Romney's party)

State	a_n	b_n
Florida	0.472222	0.527778
Virginia	0.402778	0.597222
Ohio	0.388889	0.611111
North Carolina	0.375	0.625
Colorado	0.343137	0.656863
Iowa	0.486111	0.513889
Nevada	0.465079	0.534921
Wisconsin	0.603175	0.396825
New Hampshire	0.430556	0.569444
Michigan	0.504762	0.495238

Source: http://www.270towin.com/

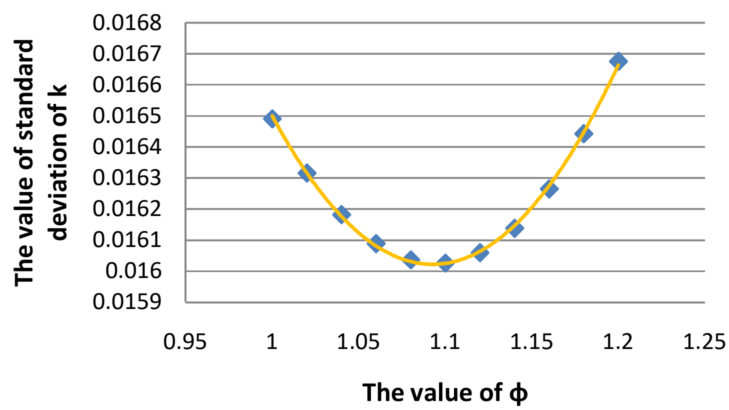
Our basic idea here is to assign φ specific values. Then, we calculated z_n for each state, and finally, k from the calculation of each state's equation. Through these calculations, we could, as a matter of fact, get 10 results of k. What we are going to do next is to calculate the mean of 10 values of k and the standard deviation of them. Because k is a constant, we wanted the standard deviation of k as small as possible. We assign φ 6 different values and compared the standard deviation of k. Table 3 and figure 1 shows the result of our comparison.

Table 3: The Value of φ and Standard Deviation of k from 10 states

The Value of φ	Standard Deviation of k from 10 states
1	0.016490141
1.02	0.016315817
1.04	0.016182088
1.06	0.016088894
1.08	0.016036685
1.1	0.016026416

1.12	0.016059556
1.14	0.016138107
1.16	0.016264634
1.18	0.016442308
1.2	0.016674967

Figure 3: The relation between standard deviation of k and "φ"



Orange line is the trend line, which is $y = 0.055x^2 - 0.121x + 0.082$ and $R^2 = 0.998$.

Thus, from the figure and table, we found out that 1.1 is the optimal value (when the standard deviation of k is at its minimum value) of φ to the nearest 0.1. As φ=1.1, k is easier to get. Table 3 is the values of k calculated from the data of 10 states.

Table 4: The values of k from the data of 10 states when φ=1.1

State	FL	VA	OH	NC	CO	IA	NV	WI	NH	MI
k	0.9980	0.9978	0.9972	0.9997	0.9960	0.9854	0.9806	0.9655	0.9483	0.9878

So far, we have worked out the mean of these ten values of k. The mean of k is 0.986. Therefore, we considered that k is a constant with the value of 0.986. (φ=1.1) However, when φ changes, k can be other values. Table 5 is to show the different values of k for different values of φ. For these values of k and φ, we could get the estimated values of final results of the presidential elections.

Table 5: The values of φ, k and γ

φ	1.02	1.04	1.06	1.08	1.1	1.12	1.14	1.16	1.18
k	0.9859	0.9859	0.9858	0.9857	0.9856	0.9855	0.9853	0.9852	0.9850
γ	0.9899	0.9927	0.9953	0.9976	0.9996	1.0014	1.0029	1.0040	1.0048

γ is the ratio of the estimated value of β (certainty to candidate A before the money was plunged) to the actual value of β from the data of 2012 U.S. presidential campaign. We found out that when φ=1.1, γ is nearest to the actual value of β.

V. Conclusion

In short, we built a mathematical model to show the optimal way of allocating a presidential candidate's financial resources under the conditions of knowing the support rate before plunging money in each state, the standard deviation of support rates of a candidate in all polls in each state and the percentage of elections' results that the party of each candidate in the history within states. Under the consideration of individual certainties within a state by normal distribution and, mathematically, probability density function; of the diminishing efficiency of the financial resources; of the money the opponent of the candidate we considered spends, of the values of constants in our models; of the law of "winner takes all" and of the mechanism of the Electoral College, we finally gave out an equation set and an inequality to optimize financial strategy in a presidential campaign for highest possibility to win the election.

Here are the strengths of our model:
1. Our model takes many real factors in different fields into consideration: normal distribution in statistics, marginal utility in economics, plurality voting system in politics, etc.. The knowledge from these subjects makes our model more comprehensive.

2. We can justify our model through the calculation of the constants in our model in the sensitivity analysis, which also means that the parameters in our model are flexible and that our model is meaningful.

3、Finally our model could be presented by an equation set and an inequality expression. In other words, our model only utilized algebraic methods, which means we can get quantitative results if we input the correct known data.

However, our model is by no means perfect. Here are the weaknesses of our model:
1. We only considered one of the two candidates, and supposed that he/her knows his/her opponent's distribution of financial resources. In fact, this is a game between two candidates. In reality, it is nearly impossible to know the exact amount of the opponent's distribution of financial resources. Game theory should be applied in this model to determine the amount of financial resources a candidate should plunge. However, it is too complex for us to include the game theory inside the model. Therefore, we ignore this game between two candidates.

2. We only referred to the data of 2012 U.S. presidential campaign; we didn't look for the regularities from more historical data.

3. We only referred to the money spent on TV advertisements in 2012 presidential campaign. Although it shows some regularity, it is by no means comprehensive.

VI. Bibliography

1. Kirby Goidel & Keith Gaddie, "*Money Matters (in Presidential Elections)*", Huffington Post, Feb. 8th, 2016.
2. https://en.wikipedia.org/wiki/Probability_density_function
3. George Casella & Roger L. Berger, "*Statistical inference*", Vol. 2. Pacific Grove, CA: Duxbury, 2002.
4. https://en.wikipedia.org/wiki/Utility
5. https://en.wikipedia.org/wiki/Marginal_utility
6. https://en.wikipedia.org/wiki/Plurality_voting_system
7. https://en.wikipedia.org/wiki/Electoral_College_(United_States)
8. *Mad Money: TV ads in the 2012 presidential campaign*, Washington Post http://www.washingtonpost.com/wp-srv/special/politics/track-presidential-campaign-ads-2012/
9. *2012 Florida/Virginia/Ohio/North Carolina/Colorado/Iowa/Nevada/Wisconsin/New Hampshire/Michigan President: Romney vs. Obama*, Huffpost Pollster
10. http://www.270towin.com/
11. http://aceproject.org/epic-en/CDTable?question=ES005

AoCMM 2016 - #822

Problem 2

Ambulance-route arrangement

Summary

Confronting the reality that the responsive times of ambulances in Manhattan are too slow, we built a model to optimize the routes of ambulances to shorten the time taken for the ambulances to arrive at crash sites and rush back to hospitals, found out the optimal distribution of ambulances among hospitals and analyzed the defects of the location of hospitals in Manhattan for emergencies in consideration of distance, traffic jams, one-way street, etc.. To achieve these goals with certain considerations, we basically applied dijkstra's algorithm, a mathematical model, to look for the shortest path within a graph.

Our team created 2091 nodes to represent important intersections in Manhattan. Subsequently, our team judged the accessible nodes for each node by mathematical inductions, calculated the distances between all two accessible nodes by geographical distance measuring, and lengthened the distance between two points by multiplying the corresponding coefficients of traffic situation respectively.

In addition to these steps, our team found the nearest nodes of each meaningful accident. (A meaningful accident means an accident that needs ambulances to rescue.) We also found out the nearest intersections of all hospitals in Manhattan. By neglecting the distance between nodes and accident scenes or hospitals that are too small to calculate, our team simplified the problem of arranging ambulance-routes into finding out the nearest path between designed nodes on a graph where distances between every two adjacent and accessible nodes are weighted by coefficients about congestion.

Finally, we gave out the methods to solve or calculated the responsive times of ambulances, how the hospitals and emergency stations in Manhattan distribute their ambulances, and some advice to New York City Government to optimize the arrangement or ambulances in the long-term.

For purpose of understanding the effect of different factors and whether our assumptions were reasonable, our team calculated several amounts of the real times spent from one intersection to another according to Google Map and what we obtained from our conclusion. By comparing these two answers, we endeavored to show that how usefulness our model is to deal with the real situations in Manhattan.

We endeavored to show more about our methodology than the technology we utilized. Although Google Map can solve this problem easily by visualization, we chose to build models by ourselves. It is more challenging, but it could be more generally used without too technical help. In short, it is important that our method, shortening responsive times by models, lowers the threshold of solving this kind of questions.

I. Introduction and Problem Restatement

1.1 Introduction

According to the question, Manhattan alone experiences over 50,000 vehicle collisions every year, 14.2% of which required ambulances to rescue. However, ambulances in New York sometimes needs around nine minutes to crash sites. We are now living in a world where efficiency matters the most; in other words, efficiency is of paramount importance, and this is particularly true when saving human beings' lives from emergencies like traffic accidents. Nine minutes of the responsive time are too long for dying people at accident scenes to wait!

A method that has the optimum efficiency is always likely to stand out from all the other manual methods. In fact, Google Map, based on satellites and manual works, is already outstanding for its system of figuring out the fastest way from a hospital or emergency station to the accident scene and back to the hospital. By using Google Map, the optimal route can be figured out within seconds. If we directly made use of Google Map, this question would be meaningless. Thus, here we considered that we cannot get help from Google Map directly except obtaining necessary data about typical traffic, which is an important factor that cannot be considered without using Google Map typical traffic system.

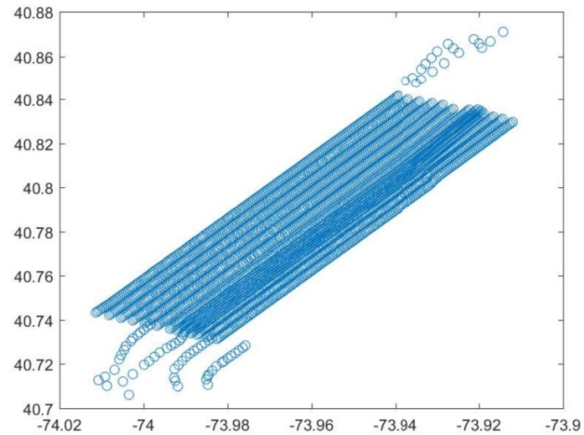
Without the direct help of Google Map, we built mathematical models and simulation models to analyze the optimum route arrangement for ambulances in order to confront similar but various situations. Our model is going to help ambulances figure out the best hospitals or emergency stations in Manhattan to launch an ambulance (or maybe ambulances) so that the rescuing vehicle(s) can reach the targeted crash sites within the least necessary time. Additionally, we can also design the route for ambulances to come back to hospitals by using the same method.

New York City is the largest city in the world with complex traffic system. To make our model more applicable to reality, we considered traffic jams, one-way or two-way streets, distances and other factors. We believe that our model can effectively help New York City Government establish a better mechanism as an alternative of Google Map when facing emergencies in the streets or on the roads in Manhattan.

1.2 Problem Restatement

Considered that the average speed of ambulance on a road without traffic jams is a constant, we wanted to find out the fastest route from one of the hospitals or emergency stations in Manhattan to crash sites and return any hospital. Therefore, we changed the map of Manhattan into a graph with nodes representing intersections and looked for the best route between two nodes in the graph. In order to take traffic jams and one-way or two-way roads into account, we assigned values to straight lines between adjacent nodes. Then our goal is to calculate the possible minimum of the sum of values on all the straight lines that an ambulance can pass and figure out the route subsequently.

Figure 1-Raw Graph of Manhattan's Intersections



X-axis represents latitudes while Y-axis represents longitudes. The graph our team created is based on Manhattan maps. Due to the convenience of calculation, we created some intersections that doesn't exist and eliminated them by mathematical methods. For purpose of conciseness, we also neglect some unnecessary points. Following passage will explain more about our plotting.

II. Definitions, Assumptions, Variables and Justifications

2.1 Definitions

Effective accident: an accident that requires ambulances to crash sites and save the casualties

Dijkstra's Algorithm: an algorithm for finding the shortest paths between nodes in a graph, which may represent, for example, road networks. It was conceived by computer scientist Edsger W. Dijkstra in 1956 and published three years later. (Dijkstra, Edsger W., 1959)

The coefficient of traffic: the coefficient representing the multiplied index of the distance between two adjacent and accessible points. For instance, if the distance between two adjacent and accessible points is 1, the coefficient of traffic is 2(Google Map typical traffic shows red of this road.), the value we set up for the straight line between these two point would be 1*2=2.

2.2 Assumptions and Justifications

In order to streamline our model, we have made several key assumptions.

Assumption： Ambulances ignore traffic lights while driving in Manhattan.
Justification: According to regulations of New York City, (Section 1104, Article 23, *The New York State Vehicle and Traffic Law*) ambulances do not have to consider the existence of traffic lights. Besides, the negligence of traffic lights makes sure that ambulances pass the intersections at its fastest speed.

Assumption: Ambulances never use roads that are along the boundary of Manhattan, such as FDR drive,
Justification: Boundary roads are always crowded according to Google Map Typical Traffic; Since all hospitals are inside Manhattan, boundary roads will definitely not be the most efficient ones. When it is in non-rush hours, it is especially no need for ambulances to go drive on those boundary roads during nighttime since there will always be more convenient and efficient routes in the inner part of Manhattan.

Assumption: The hospitals, emergency stations and accident scenes are treated as points in the map or nodes in the graph.
Justification: In comparison to the area of Manhattan Island, these places can be ignored both on the map and on the graph. Although uncertainty will appear, this assumption can maximize the simplicity and efficiency of modeling.

Assumption: The distance between a pair of adjacent intersections will mostly be measured as straight distance.
Justification: Almost every road in New York is straight, and by assuming this, the model is simplified and easier to calculate.

Assumption: A few oblique roads will not be taken into consideration when planning optimal routes.
Justification: It is hard for us to consider the adjacent nodes of a node on an oblique road in Manhattan.

Assumption: The extent of traffic jams are considered only in the unit of areas.
Justification: Since different distinct of New York City have different function, their characteristics in a specific district are similar.

Assumption: The differences of traffic among different time periods within a week are only divided into two situations: rush-hour and non-rush-hour.
Justification: The traffic situations within rush hours are quite similar. So are the traffic situations within non-rush hours.

Assumption: Green in Google Map typical traffic means that the coefficient of traffic is 1; Orange means 1.5; Red means 2; Dark Red means 2.5. The coefficient of traffic within a unit of traffic (area) is a mean of the coefficients of traffic of all the roads in

the specifically area.
Justification: Since Google Map typical traffic is based on estimation, using the average is better to measure the crowdedness of a district.

Assumption: Ambulances do not cross district to rescue the accident scene.
Justification: Since several hospitals are just located at the boundaries evenly between Mid-district and Upper-district and between Mid-district and Lower-district, the ambulances will not go to the hospitals in another district. When an ambulance is about to another district in reality, it can always find a nearer hospital in the same district because of our demarcation of district in Manhattan.

Assumption: The speed of an ambulance on a non-crowded (non-congestion; green) road is 36KM/h.
Justification: Even if the road is green in Google Map typical traffic, ambulances cannot drive faster than 36KM/h for the sake of the safety of the casualties on ambulances.

Assumption: An ambulance can only carry one casualty.
Justification: The space of ambulance is limited that only one casualty can be carried.

2.3 Variables

A is the matrix for showing which pair of points are adjacent and accessible in Mid-district of Manhattan. If two random adjacent points are accessible, we evaluated "1"; if not, we evaluate "0". A(i,j) is the number of row i column j in matrix A. If A(i,j)=1, point i and point j are adjacent and accessible; if A(i,j)=0, point i and point j are either not adjacent or inaccessible. The size of A is 2015*2015.

B is the matrix for showing the distance between every two points in Mid-district. B(i,j) equals the distance from point i to point j. The size of A is 2015*2015.

C is the matrix for showing the traffic. The numbers in each row are the same because we considered that the coefficient of traffic is a constant starting from the same intersection. C(i,j) represents the coefficient of traffic of point i.

D is the matrix of the dot product of matrix A, matrix B and matrix C. D=A.*B.*C. N(Hn) represents the amount of ambulances a hospital in Manhattan should possess for emergencies on roads.

III. Mathematical Models

3.1 Dividing Manhattan into three parts

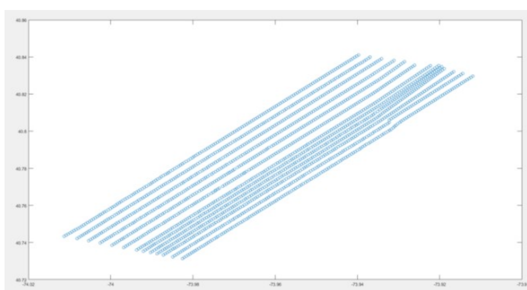
When looking at the map of Manhattan, you can easily find out that roads in some areas of Manhattan are arranged regularly. Further, these roads can be treated as the boundaries of rectangles. Therefore, we would like to pick these regularities out to use a relatively easier way with simpler mathematical models. According to my observation, roads from 14th Street to 168th Street are straight, and most of them are parallel or perpendicular to each other. We took this area out can call it Mid-district of Manhattan.

For other areas, we naturally named them as Upper-district of Manhattan for areas north to 168th Street and Lower-district of Manhattan for areas south to 14th Street. Here we can see that the boundaries between districts are 14th Street and 168 Street respectively. When we consider the latitudes and longitudes, 14th Street and 168 Street can be represented as $y = -2.385065886x + 23.16459363$ and $y = -2.364661654x + 22.63564056$. Y here is the longitude while X is the latitude.

3.2 Modeling Mid-district of Manhattan

We wanted to use a regular geometric shape to represent streets and avenues in Mid-district, which would be easier for us to know the adjacent points of a particular point. Thus, we extended all the roads in Mid-district and created many intersections that doesn't exist, such as point P1 representing the intersection of the First Avenue and 168th Street, which doesn't exist. By doing so, we created a 13*155 lattice model for Mid-District of Manhattan.

Figure 2-Raw Graph of Manhattan's Intersections in Mid-District (Lattice Model)



We divided three factors - which pair of points are adjacent and accessible, distance between every two points, the coefficients of traffic jams between every two points - into three matrices. Then we calculated the dot product of three matrices, which is the graph of Dijkstra's algorithm. By solving the function, we knew the fastest ways for ambulance to reach all accidents in Mid-district of Manhattan.

3.2.1 Access

Firstly, we built the matrix A for showing which pair of points are adjacent and accessible. Because Mid-district is represented by a lattice model, it is easy for us to find the four adjacent points. We suppose the point in the first column from the right side and the first row from up to down point P1; we suppose the point in the first column from the right side and the last row from up to down P155; we suppose the point in the last column from the right side and the first row from up to down P1861(1861=1+(13-1)*155); we suppose the point in the last column from the right side and the last row from up to down P2015(2015=13*155). Then, for point P_h (h is an integer ranging from 1 to 2015), its adjacent points are P_{h-1}, P_{h+1}, P_{h+155}, P_{h-155}。 Of course h-1, h+1, h+155 and h-155 should be within the range of [1, 2015]. If the number is out of the range, then we should definitely say that the point does not exist.

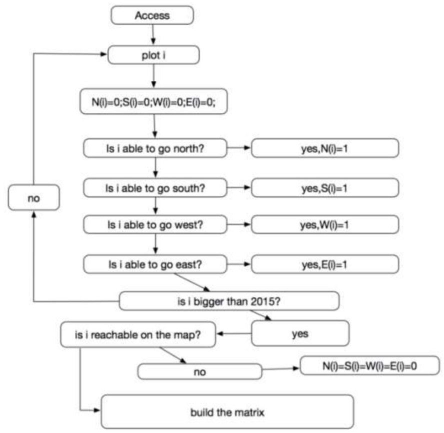
To achieve the matrix A from the perspective of access, we actually created a 4*2015 table first. Basically, it shows whether a point can go to the upper point, lower point, point on the right side and point on the left side by creating four functions N(i), S(i), W(i), E(i) respectively. If a point can access to the adjacent point of its left side, for instance, W(i)=1; otherwise, W(i)=0. So do N(i), S(i), and E(i).

With this 4*2015 table, we then evaluated the matrix A much more easier. Matrix A basically would like to indicate that when point X and point Y are adjacent and point X can access to point Y but point Y cannot access to point X, the vector from point X to point Y is evaluated "1"; the vector from point Y to point X is evaluated "0". In matrix A, the information is given by that "1" at the row of X and column of Y while "0" at the row of Y and column of X. Mathematical expression are that A(X,Y)=1 and A(Y,X)=0. By repeating this operation to all points in Mid-district (2015 points) , we eventually worked out a 2015*2015 matrix.

Actually, there are many points that do not exist such as aforementioned point P1. To these points, we just evaluated "0". For example, all the numbers in row 1 was evaluated zero (A(1,n)=0, $n \in [1, 2015]$) because the intersection P1 represents doesn't exist in reality. Through this operation, we eliminated the ineffectual nodes.

Also, we found out many regularities in New York road system, like most of the one-way streets or avenues to east or north are next to one-way streets or avenues to west or south. We applied these laws into consideration, which simplified our models, but we still adjusted something automatically generated from the regularities because there are some exceptions of these regularities in Manhattan.

Figure 3-Flow Diagram of Access



3.2.2 Distance

Secondly, we created the matrix B of distance. To this end, we acquired the location of every node on the graph of Mid-district of Manhattan. Because the lattice model is consist of 13*155=2015 nodes, and streets from 14th street to 168 street are parallel in nearly same distance. Therefore, we actually created 13 lines from 168th street in the north to 14th street in the south and divided each line into 154 identical lines with same length by using latitude and longitude. By this operation, we obtained the location of every node on the graph of Mid-district of Manhattan in a relatively easier and more accurate way.

After acquiring the locations of points, we can thus calculate the distance. According to geographical distance, the formula for distance when the latitudes and longitudes of two points are given is that:

Let the location of Point X (LonX, LatX), that of Point Y (LonY, LatY). For the point north to equator we define $\mathrm{MLatX} = 90° - \mathrm{LatX}$; $\mathrm{MLatY} = 90° - \mathrm{LatY}$. Then two new location points would be (LonX, MLatX) and (LonY, MLatY). According to trigonometry, the following formulas can be:

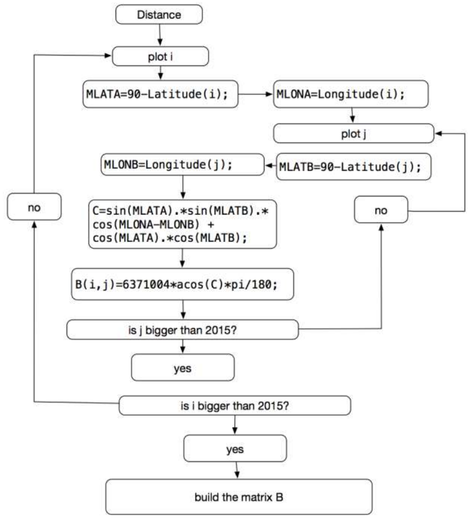
$$C = \sin(\mathrm{MLatX}) \times \sin(\mathrm{MLatY}) \times \cos(\mathrm{LonX} - \mathrm{LonY}) + \cos(\mathrm{MLatX}) \times \cos(\mathrm{MLatY})$$

$$D_{XY} = D_{YX} = R \times \mathrm{Arccos}(C) \times \frac{\pi}{180°}$$

D_{XY} is the distance between node X and node Y; R is the radius of earth, which is 6371004 meters.

In matrix B, D_{XY} is the value at X row Y column (B(X,Y)= D_{XY}) while D_{YX} is the value at Y row X column (B(Y,X)= D_{YX}).

Figure 4-Flow Diagram of Distance



3.3.3 Congestion

Thirdly, we created the matrix C of the coefficients of traffic. To this end, we referred typical traffic in Google Map and divided Mid-district of Manhattan into 18 sub-districts. We took the average of the coefficients of traffic in a sub-district as a unit when considering the traffic jams. Although Green represents 1, Orange represents 1.5, Red represents 2, Dark Red represents 2.5, the average values could be 1, 1.25, 1.5, 1.75, 2, 2.25, 2.5 as a result that the roads with different conditions within a sub-district are mixed.

Time is a great factor in traffic. We distinguished time of a week by taking into account of rush hours and non-rush hours. Rush hour is 9 a.m. to 7 p.m. on weekdays and 11 a.m. to 9 p.m. at weekends, and the rest of the time in a week is considered as non-rush hours. For two different times, we created two tables to show the average of the coefficients of traffic within a sub-district.

To be clear, let us think about nodes in zone 13 as example (zone 13 is a sub-district in Mid-district of Manhattan). It is around 5 p.m., so it is definitely rush-hours. Then we can obtain the average of the coefficients from tables. In this case, the average of the coefficients of traffic in zone 13 in rush-hours is 1.5. Then all the points (for instance, point Pn to Pn+k) within this sub-district would be valued as 1.5. In matrix C, all the numbers in rows from n to n+k would be marked as 1.5. (For $t \in [1, 2015], t \in Z^+,\ q \in [n, n+k], q \in Z^+, C(q,t) = 1.5$)

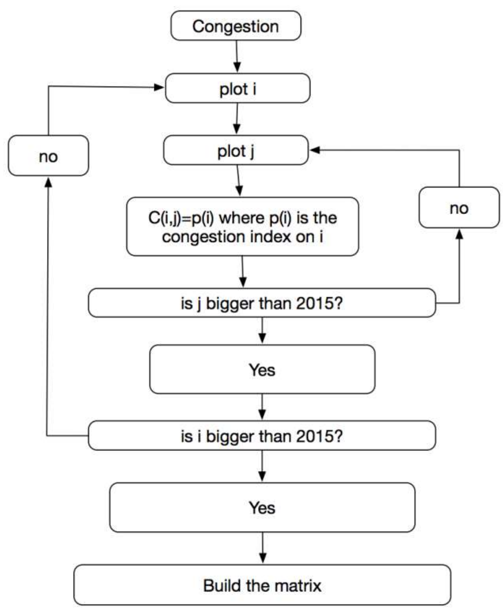
Table 1-The average of coefficients of traffic within sub-districts in Mid-district of Manhattan

	Range of n in P_n	Rush-Hour	Non-rush-hour
Zone M1	n∈[155k-154,155k-121], k∈[1,7]	1	1
Zone M2	n∈[155k-154,155k-121], k∈[8,9]	1.25	1

Zone M3	n∈[155k-154,155k-121], k∈[10,13]	1.25	1
Zone M4	n∈[155k-120,155k-96], k∈[1,7]	1.25	1
Zone M5	n∈[155k-120,155k-96], k∈[8,9]	1.25	1
Zone M6	n∈[155k-120,155k-96], k∈[10,13]	1.5	1
Zone M7	n∈[155k-95,155k-72], k∈[1,7]	1.25	1
Zone M8	n∈[155k-95,155k-72], k∈[8,9]	1	1
Zone M9	n∈[155k-95,155k-72], k∈[10,13]	1.25	1
Zone M10	n∈[155k-71,155k-46], k∈[1,7]	1.5	1
Zone M11	n∈[155k-71,155k-46], k∈[8,9]	1	1
Zone M12	n∈[155k-71,155k-46], k∈[10,13]	1.25	1
Zone M13	n∈[155k-45,155k-20], k∈[1,6]	1.5	1.25
Zone M14	n∈[155k-45,155k-20], k∈[7,9]	1.75	1.25
Zone M15	n∈[155k-45,155k-20], k∈[10,13]	1.5	1.25
Zone M16	n∈[155k-19,155k], k∈[1,6]	1.5	1.25
Zone M17	n∈[155k-19,155k], k∈[7,9]	1.75	1.25
Zone M18	n∈[155k-19,155k], k∈[10,13]	1.25	1

Source: Google Map typical traffic

Figure 5-Flow Diagram of Congestion



After obtaining these three matrices: matrix A, matrix B and matrix C, we multiplied them together to acquire their dot product. Let us name the final matrix be D, D=A.*B.*C. We used Dijkstra's Shortest-Path Algorithm to find out the fastest routes for accidents. We input D as the graph, all the nodes which represent hospitals and emergency stations in Manhattan (we named it array s(i)), and the location of a particular accident scene (we named it "d"). Function Dijkstra(D, s(i), d) will basically obtain a minimum sum of values between every two points we assigned. After gathering nineteen minimum sums from hospital s(1) to s(19), we compared them by programming and find out the smallest one. If the smallest one is from s(2), it means that hospital 2 should go for the rescue of the accident and Dijkstra(D, s(2), d) can also work out the route of the ambulance by listing all the numbers of nodes the ambulance passes.

The programming of find out the results for all accidents is too big to calculate. Still, we worked out some of them, and the results are compared with the results from special-case-model in sensitivity analysis. In addition, we worked out the distribution of ambulances by finding out the amount of accidents a hospital should responsive because if its ambulances response, the accident can be solved fastest. For every accident scene, we found out the optimal hospital (such as H01). Then N(H01)=N(H01)+1. By this operation, we calculated the numbers of all hospitals. Subsequently, we divided the total number of ambulances by them and got the percentage of the total number of ambulances that a hospital should have to minimize the responsive time.

3.3 More accurate case inside Mid-district

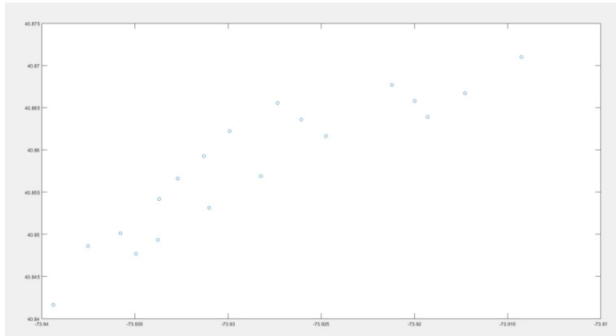
In fact, before building the lattice model for whole Mid-district of Manhattan, we tested our method through a smaller region. We picked the region east to the central park where starts from 67th street and ends at 97th street. Similarly, we calculated the dot product of access matrix, distance matrix and congestion matrix. Then we made use of dijkstra's algorithm to help the hospitals in this region find the accident scenes and the route for return. This model will be used for sensitivity analysis and a comparison about accuracy with the whole Mid-district case.

3.4 Upper-district and Lower-district Analysis

Basically, we treated upper-district and lower-district of Manhattan the same way as we dealt with Mid-district, but there are some slight differences.

In the upper-district case, we created 19 nodes for its graph, and find out the adjacent and accessible nodes for each node manually. For the matrix of distance and that of congestion (we consider the upper district as one sub-district of traffic), the method remains. (The rush hour's congestion coefficient is 1.5 while the non-rush hour's is 1.)

Figure 6-Raw Graph of Manhattan's Intersections in Upper-District



In the lower-district case, we created 57 nodes for its graph, and we named them Q1~Q57. Q1~Q56 is a 4*14 lattice model. In this case, we eliminated the consideration of one-way street by conceiving that two one-way streets with opposite direction as one two-way street. By this operation, the model is simplified. Q57 is a node between Q28 and Q42. Q28 cannot direct connect to Q42 because of the reality, so we created Q57. Q28 and Q57 can connect to each other while Q42 and Q57 can connect to each other, too. For the distance matrix, we repeated the procedures in the Mid-District case. For the congestion matrix, we divided the lower-district into four traffic sub-districts and consider their level of traffic respectively like what we did in the Mid-district case.

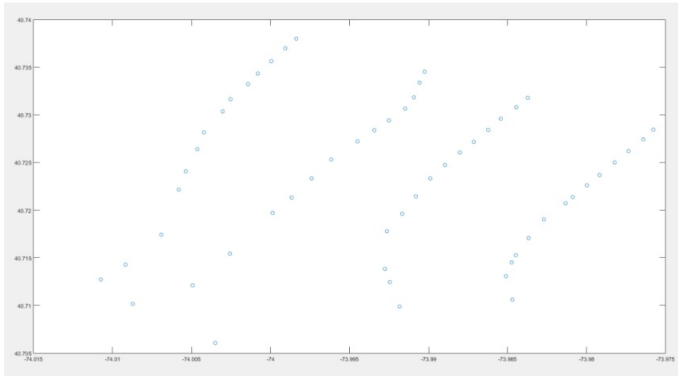
With the help of matrices, we again applied dijkstra's algorithm and finally came out the result that for each accident, which route the hospital or emergency station should choose to crash sites and return in the shortest responsive time like what we did in the case of Mid-district of Manhattan.

Table 2-The average of coefficients of traffic within sub-districts in Lower-district of Manhattan

	Range of n in Qn	Rush Hour	Non-rush Hour
Zone D1	n∈[14k-13,14k-4], k∈[1,2]	1.25	1
Zone D2	n∈[14k-13,14k-4], k∈[3,4]	1.5	1.25
Zone D3	n∈[14k-3,14k],k∈[1,2]	1.5	1.25
Zone D4	n∈{57}∪[14k-3,14k],k∈[3,4]	1.5	1.25

Source: Google Map typical traffic

Figure 7-Raw Graph of Manhattan's Intersections in Lower-District



IV. Sensitivity Analysis

As shown, we divided the whole Manhattan into twenty-three different sub-districts and only concerned the congestion within a sub-district. However, it is by no means perfectly accurate. We built a model for a more accurate case (Let's call it "special case") inside Mid-district as we mentioned. The region starts from 67th street and ends at 97th street, and is east to the central park. In this special case, we considered the coefficients of congestion of roads between every two adjacent and accessible points. Therefore, by comparing using dijkstra's algorithm on the lattice model of whole Mid-district and using dijkstra for this special case, we can have a better understanding of how the accuracy of coefficients of congestion affect the results.

Table 3-Comparison about responsive time by non-special case model and special case model

Model	Accident Location	To which hospital	Responsive time to accident scene	Returning time to hospital
Non-special case	(40.7794367, -73.975068)	H14	1.411908333	1.420123333
Special case			1.816666667	1.433333333
Non-special case	(40.7788219, -73.953985)	H13	1.844	1.8485
Special case			1.7	1.7
Non-special case	(40.7768682, -73.9554089)	H13	1.435985	1.425386667
Special case			1.966666667	2.358333333
Non-special case	(40.7740216, -73.9545587)	H11	1.275355	0.995853333
Special case		From H13 to H11	1.45	1.083333333

The unit of time here is minute.

Table 4-Comparison about Ambulance's route by non-special case model and special

case model

Model	Accident Location	To which hospital	Crash-site-direction	Returning Direction
Non-special case	(40.7794367, -73.975068)	H14	Go straight along 2nd Avenue to E 90th Street and turn left; go straight to 1st Avenue	Go straight along 1st avenue to E 97th Street and turn left
Special case			Same to non-special	Same to non-special
Non-special case	(40.7788219, -73.953985)	H13	Go straight along Park Avenue to E 82nd Street and turn right; go straight to 3rd Avenue and turn right; go straight to E 86th Street	Go straight along E 86th Street to Lexington Avenue and turn left; go straight to E 77th Street and turn right; go straight to Park Avenue
Special case			Same to non-special	Go straight along E 86th Street to Park Avenue and turn left; go straight to E 77th Street
Non-special case	(40.7768682, -73.9554089)	H13	Go straight along Park Avenue to E 82nd Street and turn right; go straight to 3rd Avenue and turn right; go straight to E 83th Street	Go straight along E 83th Street to Lexington Avenue and turn left; go straight to E 77th Street and turn right; go straight to Park Avenue
Special case			Same to non-special	Same to non-special
Non-special case	(40.7740216, -73.9545587)	H15	Go straight along 1st Avenue to E 80th Street and turn left; go straight to 2nd Avenue	Go straight 2nd avenue to E 74th Street and turn left; go straight to 1st avenue
Special case		From H13 to H11	Go straight along Park Avenue to E 79th Street and turn right; go straight to 2nd Avenue and turn left; go straight to E 80th Street	Same to non-special

Here we can conclude several points from the sensitivity analysis:

• The differences of responsive times are small; the routes of ambulances are mostly identical. Special-case model is absolutely accurate. By this analysis, we can also know that non-special-case model is also accurate.

•New York is a crowded city; the coefficient of congestion of every road between two accessible and adjacent nodes should be taken into account in order to make the model more accurate.

• Considering whether roads are one-way or two-way is vital; we did consider it even in the non-special case. However, from these data, we found out that actually the situation that an ambulance crashes the site from a hospital and return to another one does exist!

• Although there are differences in results between non-special model and special model, we can still say that both of them are accurate enough to calculate the responsive time.

V. Conclusion

So far, we built three matrices in each district of Manhattan. Three matrices concern whether two random nodes are adjacent and accessible, the distance between every two points, the coefficient of congestion respectively. By calculating their dot product, we got the graph for dijkstra's algorithm. Then we input the locations of all accidents and the locations of all 24 hospitals. Dijkstra's algorithm helps us to calculate the smallest responsive time from which hospital to crash the site and should return to which hospital and list the route. Then we got all responsive times and routes for all accidents theoretically. By further calculation of N(Hn), we can know how ambulances distribute in the hospitals in Manhattan.

Our ideas of how to solve this question evolves a lot from the beginning to the answers we are writing now. At first, we only considered that the distances of the hospitals and the accident scenes matter regardless of actual distance, congestion, one-way or two-way, etc. However, we found out that the results are by no means accurate. For example, the distance between (40.7794367, -73.975068) and H14 is 640 meters, but the actual distance is around 900 meters. However, we still do not think that distance is the only factor, so we added more and more factors into consideration. We built matrices about these factors and calculated the dot product of them. Finally, by using dijkstra's algorithm, we found out the fastest path from a hospital to crash the site and return back to a hospital. In addition, we also considered the situation that an ambulance departs from and arrives at different hospitals from sensitivity analysis.

Nevertheless, to be frank, even we considered many factors in our models, it is still not very accurate. Google Map has actually already solved this kind of question for a long time. It can already get the results directly by inputting the location of hospitals and accident scenes with considerations more than what we can think of. Google Map's typical traffic is also very strong to deal with congestion. However, we still believe that our model is meaningful, because we conducted several calculations like the distribution of ambulances among hospitals in Manhattan that Google Map cannot directly deal with based on a relatively simple model. By not directly referring Google Map, we made our model more applicable to graphs or maps that may not as well-functional as Google Map but still need to deal with for useful mathematical operations.

By calculation, we also found out that the average responsive time for accidents in Midtown like places around Times Square is around 4.5 minutes; however, the average responsive time for accidents in whole Manhattan is about 1.5 minutes. By checking the map, we can also find out that there are inadequate hospitals around Midtown like places around Times Square. Our suggestion to the mayor of New York City is to build more hospitals or emergency stations in Midtown to shorten the responsive time at that area.

Here are the strengths of our model:
1. We took many factors-whether roads are one-way or two-way, congestion, etc.-in account, making our model precise and accurate.
2. Our method can be generalized in the cases of all cities and towns throughout the world.
3. We considered some special factors and cases, such as that an ambulance from Hospital A may crash the site and return Hospital B instead of going back to Hospital A because in reality returning to B may be the fastest way.

Here are the shortages of our model:
1. When we established the lattice model of Mid-district of Manhattan and the nodes in Upper-district and Lower-district in Manhattan for intersections, we spent too much time doing the manual work.
2. There are too many input data in our model that even computer calculated for a long time. For the whole case, the computer actually cannot work out the result.
3. We ignored the boundary roads of Manhattan, which may become the fastest paths for rescuing accidents.
4. There might be special congestions due to emergent accidents on streets which make these roads especially crowded. However, we didn't take it into account.
5. Our coefficients of congestion do not very accurately reflect the reality. If we have more time, we should change coefficients into bigger ones to rebuild the matrices of congestion.

VI. Bibliography

1. *The New York State Vehicle and Traffic Law*
2. Dijkstra, Edsger W. "*A note on two problems in connexion with graphs.*" Numerische mathematik 1.1 (1959): 269-271.
3. More information about Google Map Typical Traffic: https://support.google.com/maps/answer/3092439?co=GENIE.Platform%3DDesktop&hl=en

Appendix

Appendix for question 1:

[Code]: Maximize $W_A = \sum_{n=1}^{50} V(n)$ known that:

$$
\begin{cases}
\sum_{n=1}^{50} M_n = t \\
z_n = a_n{}^{\varphi} \times M_n - b_n{}^{\varphi} \times Q_n \\
K(z_n) = -(1-\beta)k^{z_n} + 1, k \in (0,1) \\
T(\beta)_n = \int_{\beta-0.05\%}^{\beta+0.05\%} \frac{1}{\sigma_n\sqrt{2\pi}} e^{\frac{-(\beta-\mu_n)^2}{2\sigma_n{}^2}} d\beta \\
G(T(\beta)_n) = \begin{cases} T(\beta)_n, K'(z_n) \in [0.5,1) \\ 0, K'(z_n) \in (0,0.5) \end{cases} \\
V(n) = \begin{cases} e(n), & \int_0^1 G(T(\beta)_n)\, d\beta \in (0, 0.5) \\ 0, & \int_0^1 G(T(\beta)_n)\, d\beta \in (0.5, 1) \end{cases}
\end{cases}
$$

```
Z=input('the number of voters in each state in electoral college based
on the alphabetical order=')
% Z is a 1*50 matrix
Y=0;
% Y acted as a base to test whether the new trial is a better allocation
scenario or not
for i=1:10000
% This program aims to choose the best allocation scenario among 10000
data
    s=0;
    for j=1:50
        A(j)=rand;       % Randomly construct the polls in 50 states
        s=s+A(j);        % find the sum of these 50 polls
    end
    for j=1:50
        M(j)=round(A(j).*R/s);
        % suppose the candidate has total money R(unit:millions)
        % calculate the weight of each state's allocation
    end
    for o=1:50
        % calculate the final result of each state based on the data
        z=a.*M(o)-b.*Q;
        % Q indicates the amount of money the opponent spent on same state
        % Q is the given number
        x0=1-0.5/(k.^z);
```

```
        x=x0:0.000001:1;
        D=StandardDeviation;
        M=preObama;
        w=0;
        y=1/(D(o)*(2*3.1415926).^0.5)*exp((-(x-M(o)).^2)/(2*D(o).^2));
        s(o)=trapz(x,y);
        if s(o)>=0.5
        % to decided the result based on the 'plurality voting system'
            b(o)=Z(o);
        else
            b(o)=0;
        end
        w=w+b(o);
    end
    if w>Y,
        % the bigger w, the higher the probability the candidate wins
        Y=w;
        for q=1:50
            H(q)=A(q)/10000;
        end
    end
end
xlswrite('allocation_scenario.xlsx',H)
```

Appendix for question 2:

[Table] The locations of hospitals in Manhattan

Hospital Code	LATITUDE	LONGITUDE
H01	40.73333	-73.9823
H02	40.7348	-73.9896
H03	40.73707	-73.9768
H04	40.7377	-74.0009
H05	40.73912	-73.9753
H06	40.74184	-73.9746
H07	40.7476	-74.005
H08	40.76285	-73.9558
H09	40.76396	-73.9632
H10	40.76465	-73.954
H11	40.76967	-73.9529
H12	40.7698	-73.9871
H13	40.77387	-73.9607
H14	40.78508	-73.9451
H15	40.78958	-73.9702
H16	40.78968	-73.9533

H17	40.80561	-73.9613
H18	40.81414	-73.9397
H19	40.84174	-73.9418
H20	40.87331	-73.913
H21	40.71001	-73.9862
H22	40.71039	-74.0051
H23	40.71815	-73.9996
H24	40.72152	-73.9959

[Code] Dijkstra's algorithm for the whole case of Manhattan given that it is rush-hour or non-rush-hour

```
G=finalsolution;
% choose the appropriate matrix which is able to indicate the road condition
for v=1:24597
    % include the accident plot needed to be calculated
if
(LONGITUDE>=-2.385065886*LATITUDE+23.16459363)&&(LONGITUDE<=-2.364661
654*LATITUDE+22.63564056)
    % judge whether the plot of accident(v) is in the midtown or not
    % // the start to find the nearest crossroad
    k1=10000;
    MLATA=90-LATITUDE_accident(v);
    MLONA=LONGITUDE_accident(v);
    for m=1:2015
        MLATB=90-Latitude(m);
        MLONB=Longitude(m);
        if v~=m
            C = sin(MLATA).*sin(MLATB).*cos(MLONA-MLONB) +
cos(MLATA).*cos(MLATB);
            a=6371004*acos(C)*pi/180;
            if a<k1
                k1=a;n=m;
            end
        end
    end
    i=n;
    % the end to find the nearest crossroad//
    k(1,i)=10000000;
    % construct the block to testify the best route
    if
(i>=1&&i<=43)||(i>=976&&i<=978)||(i>=156&&i<=195)||(i>=311&&i<=350)||
(i>=466&&i<=508)||(i>=621&&i<=658)||(i>=776&&i<=805)||(i>=931&&i<=960
)||(i>=1086&&i<=1109)||(i>=1241&&i<=1253)||(i>=1396&&i<=1408)||(i>=15
```

```
51&&i<=1608)||(i>=2009&&i<=2015)||(i>=922&&i<=924)||(i>=1145&&i<=1156
)||(i>=1158&&i<=1167)||(i>=1170&&i<=1174)||(i>=1176&&i<=1188)||(i>=11
90&&i<=1194)||(i>=1300&&i<=1311)||(i>=1313&&i<=1322)||(i>=1324&&i<=13
29)||(i>=1331&&i<=1343)||(i>=1345&&i<=1349)
        k(1,i)=0;
        f1(1,i)=0;
        f2(1,i)=0;
        % eliminate the plot out of the map
    else
        for j=1:19
            m=Hospitalplot(j);
            % decide the hospital plot #1
            [a L1(i)] = dijkstra(G,m,i);
            % calculate the route and total distance #1
            for u=1:19
                n=Hospitalplot(u);
                % decide the hospital plot #2
                [b L2(i)] = dijkstra(G,i,n);
                % calculate the route and total distance #2
                e=a+b;
                % figure out the total back-and-forth distance
                if e<k(1,i)
                    % compare the new total distance with the best route
previously
                    A(i)=a;
                    % the distance from starting hospital to accident
                    B(i)=b;
                    % the distance from accident to final hospital
                    Z=i;
                    k(1,i)=e;
                    % k(1,i):the final total back-and-forth distance
                    f1(1,i)=m;
                    % the plot of hospital where the ambulance departs
                    f2(1,i)=n;
                    % the plot of hospital where the ambulance arrives
                % NOTE: 'As the uncertainty that ambulance might started
                % from plot alpha but finally goes to plot beta, f1(1,i)
                % and f2(1,i) might not be identical.'
                end
            end
        end
    end
    k(1,i);
    % the shortest total back-and-forth distance
```

```
    [a L1(i)] = dijkstra(G,f1(1,i),Z);
    [b L2(i)] = dijkstra(G,Z,f2(1,i));
    L1(i);
    % the shortest route from starting hospital to accident
    L2(i);
    % the shortest route from accident to fianl hospital
    y=f1(1,i)
    T(y)=T(y)+1;
    %  # accidents needed the ambulance started as plot y
end
% NOTE: 'as this program is an example, if the plot of accident(v) is in
% downtown or uptown, the parameter (inserted matrix) of the certaon
variable should be changed.
% That is to say, for instance, this program is just suitable for midtown.
If this is
% uptown or downtown, matrix G should be altered.'
end
```

[Code] Matrix D-final matrix of Mid-district

```
D=midtown_access.*midtown_distance.*Midtown_Congestion_rush_hours;
xlswrite('midtown_final_matrix.xlsx',B)
```

[Code] Matrix A-whether two nodes are accessible and adjacent

```
for i=1:2015
    for j=1:2015
        A(i,j)=0;
    end
end
for i=1:2015
    N(i)=0;
    W(i)=0;
    S(i)=0;
    E(i)=0;
    T(i)=0;
    % manually give the zeros
    if
(i>=1&&i<=155)||(i>=311&&i<=465)||(i>=776&&i<=930)||(i>=1195&&i<=1240
)||(i>=1505&&i<=1550)||(i>=1764&&i<=1860)
        N(i)=1;
        % testify whether plot i could go north
    end
    if
(i>=156&&i<=310)||(i>=466&&i<=620)||(i>=931&&i<=1085)||(i>=1350&&i<=1
395)||(i>=1551&&i<=1705)||(i>=1985&&i<=2015)
```

```
        S(i)=1;
        % testify whether plot i could go south
    end
    if
(i>=621&&i<=755)||(i>=1086&&i<=1144)||(i>=1254&&i<=1299)||(i>=1396&&i
<=1505)||(i>=1706&&i<=1764)||(i>=1861&&i<=1955)
        N(i)=1;
        S(i)=1;
        % testify whether plot i could go both north and south
    end
    for k=1:14
        if i==155*k
            W(i)=1;
            E(i)=1;
            T(i)=1;
            W(i-9)=1;
            E(i-9)=1;
            T(i-9)=1;
            W(i-20)=1;
            E(i-20)=1;
            T(i-20)=1;
            W(i-28)=1;
            E(i-28)=1;
            T(i-28)=1;
            W(i-43)=1;
            E(i-43)=1;
            T(i-43)=1;
            W(i-58)=1;
            E(i-58)=1;
            T(i-58)=1;
            W(i-65)=1;
            E(i-65)=1;
            T(i-65)=1;
            W(i-72)=1;
            E(i-72)=1;
            T(i-72)=1;
            W(i-82)=1;
            E(i-82)=1;
            T(i-82)=1;
            W(i-92)=1;
            E(i-92)=1;
            T(i-92)=1;
            W(i-121)=1;
            E(i-121)=1;
```

```
        T(i-121)=1;
        W(i-131)=1;
        E(i-131)=1;
        T(i-131)=1;
        W(i-141)=1;
        E(i-141)=1;
        T(i-141)=1;
        W(i-152)=1;
        E(i-152)=1;
        T(i-152)=1;
        W(i-153)=1;
        E(i-153)=1;
        T(i-153)=1;
        W(i-154)=1;
        E(i-154)=1;
        T(i-154)=1;
        if k>=7&&k<=13
            W(i-96)=1;
            E(i-96)=1;
            T(i-96)=1;
        end
        if k>=1&&k<=11
            W(i-111)=1;
            E(i-111)=1;
            T(i-96)=1;
        end
        W(i-52)=0;
        T(i-52)=0;
        E(i-51)=0;
        T(i-51)=0;
        E(i-46)=0;
        T(i-46)=0;
        W(i-45)=0;
        T(i-45)=0;
        E(i-27)=0;
        T(i-27)=0;
        W(i-112)=0;
        T(i-112)=0;
        % assign the number which indicate whether a certain plot is
accessible to each direction or not
        for j=0:154
            a=i-j;
            if T(a)==0
                if mod(j,2)==0
```

```
                    E(a)=1;
                else
                    W(a)=1;
                end
                % make sure the direction of one-way street
            end
        end
    end
end
end
for i=1:2015
    if
(i>=1&&i<=43)||(i>=976&&i<=978)||(i>=156&&i<=195)||(i>=311&&i<=350)||
(i>=466&&i<=508)||(i>=621&&i<=658)||(i>=776&&i<=805)||(i>=931&&i<=960
)||(i>=1086&&i<=1109)||(i>=1241&&i<=1253)||(i>=1396&&i<=1408)||(i>=15
51&&i<=1608)||(i>=2009&&i<=2015)||(i>=922&&i<=924)||(i>=1145&&i<=1156
)||(i>=1158&&i<=1167)||(i>=1170&&i<=1174)||(i>=1176&&i<=1188)||(i>=11
90&&i<=1194)||(i>=1300&&i<=1311)||(i>=1313&&i<=1322)||(i>=1324&&i<=13
29)||(i>=1331&&i<=1343)||(i>=1345&&i<=1349)
        % eliminate the plots whihc are out of the map
        N(i)=0;
        S(i)=0;
        W(i)=0;
        E(i)=0;
    else
    % turn the accessibility into the matrix
    if (N(i)==1)&&(i-1>0)
        A(i,i-1)=1;
    end
    if (S(i)==1)&&(i<=2014)
        A(i,i+1)=1;
    end
    if (W(i)==1)&&(i+155<=2015)
        A(i,i+155)=1;
    end
    if (E(i)==1)&&(i-155>0)
        A(i,i-155)=1;
    end
    switch i
        case 1803
            A(i,1654)=1;
        case 1654
            A(i,1803)=1;
            A(i,1505)=1;
```

```
    case 1505
        A(i,1654)=1;
    case 1364
        A(i,1221)=1;
    case 1221
        A(i,1075)=1;
    case 1499
        A(i,1344)=1;
    case 1344
        A(i,1498)=1;
        A(i,1189)=1;
    case 1189
        A(i,1344)=1;
        A(i,1034)=1;
    case 1033
        A(i,1189)=1;
    case 1485
        A(i,133)=1;
    case 1330
        A(i,1485)=1;
        A(i,1175)=1;
    case 1175
        A(i,1330)=1;
        A(i,1020)=1;
    case 1020
        A(i,1175)=1;
    case 1478
        A(i,1323)=1;
    case 1323
        A(i,1478)=1;
        A(i,1168)=1;
    case 1168
        A(i,1323)=1;
        A(i,1015)=1;
    case 1014
        A(i,1168)=1;
    case 1003
        A(i,1157)=1;
    case 1157
        A(i,1003)=1;
        A(i,1312)=1;
    case 1312
        A(i,1157)=1;
        A(i,1468)=1;
```

```
        case 1468
            A(i,1312)=1;
        % plug the accessibility for detail plots
    end
    end
end
for i=1:2015
    for j=1:2015
        B(i,j)=A(i,j);
        % turn the accessibility to a complete usable matrix
    end
end
xlswrite('midtown_access.xlsx',B)
```

[Code] Matrix B-Distance Matrix of Mid-district

```
for i=1:2015
    for j=1:2015
        B(i,j)=0;
        % manually make the zeros
    end
end
for i=1:2015
    MLATA=90-Latitude(i);
    MLONA=Longitude(i);
    for j=1:2015
        MLATB=90-Latitude(j);
        MLONB=Longitude(j);
        C = sin(MLATA).*sin(MLATB).*cos(MLONA-MLONB) +
cos(MLATA).*cos(MLATB);
        Distance = 6371004*acos(C)*pi/180;
        % calculate the distance between two plots on map
        B(i,j)=Distance;
    end
end
xlswrite('midtown_distance.xlsx',B)
```

[Code] Matrix C-Matrix about coefficients of congestion in Mid-district, here we used rush-hour one as a case

```
for i=1:2015
    Z(i)=1;
    for j=1:2015
        C(i,j)=0;
    end
end
```

```
for i=1:2015
    for k=1:13
        if i==155*k
            for j=0:154
                if (k>=1&&k<=7)
                    if (j>=72&&j<=120)
                        Z(i-j)=1.25;
                        % the congestion index of previous road is 1.25
                    elseif (j>=0&&j<=71)
                        Z(i-j)=1.5;
                        % the congestion index of previous road is 1.5
                    end
                elseif (k>=8&&k<=9)
                    if (j>=96&&j<=154)
                        Z(i-j)=1.25;
                        % the congestion index of previous road is 1.25
                    elseif (j>=0&&j<=45)
                        Z(i-j)=1.75;
                        % the congestion index of previous road is 1.75
                    end
                elseif (k>=10&&k<=13)
                    if
(j>=121&&j<=154)||(j>=72&&j<=95)||(j>=46&&j<=71)||(j>=0&&j<=19)
                        Z(i-j)=1.25;
                        % the congestion index of previous road is 1.25
                    elseif (j>=96&&j<=120)||(j>=20&&j<=45)
                        Z(i-j)=1.5;
                        % the congestion index of previous road is 1.5
                    end
                end
            end
        end
    end
end
for u=1:2015
    for v=1:2015
        if Z(u)==1
            C(u,v)=1;
        elseif Z(u)==1.25
            C(u,v)=1.25;
        elseif Z(u)==1.5
            C(u,v)=1.5;
        elseif Z(u)==1.75
            C(u,v)=1.75;
```

```
        end
        % establish the matrix represents the congestion between each
certain points
    end
end
xlswrite('Midtown_Congestion_rush_hours.xlsx',C)
```

Here we omitted our code about Upper-district and Lower-district. They are similar to the Mid-district's one.