Optimizing Polling Locations Based on Public Transportations

Yueyan Chen (cyyan) Zirui Liu (liuzirui)
Young Jun Choi (yjunchoi) Yuchen Zhang (yzhang71)
CS 591 Data Mechanics – Fall 2017

Introduction

According to statistics collected by the U.S. Elections Project, about 40 percent of U.S. citizens did not participate in 2016 presidential elections. [1] Although many states provide various solutions to fix America's voter turnout problem, voter turnout rates have not increased in last two decades. With researching the relationship between voter turnout rates and the other factors, we found the journal, "Increasing Voter Turnout: Can Mass Transit Help?". [2]. Since Boston is one of the top 10 cities for public transportation, we decided to apply this journal to Boston to reconfigure 255 optimal polling locations across Boston. [3]

Datasets

We collected 4 data sets to reconfigure 255 optimal polling locations in 22 wards across Boston.

- Wards: Geospatial data for wards in Boston (https://data.boston.gov/dataset/wards)
- Polling Locations: Set of polling location coordinates in city of Boston (https://data.boston.gov/dataset/polling-locations)
- Bus Stops: Set of bus stop coordinates in city of Boston (http://datamechanics.io/data/wuhaoyu_yiran123/MBTA_Bus_Stops.geojson)
- MBTA: Set of MBTA T station coordinates in city of Boston (http://erikdemaine.org/maps/mbta/mbta.yaml)

After retrieving each dataset and storing it in our database (MongoDB), we divided polling locations, bus stops, and MBTA T stations into 22 wards based on their coordinates. (pollingLocation.py, bus_by_ward.py, MBTA_by_ward.py) During research, we used these new assembled datasets to optimize polling locations in Boston.

Methodology

We used a k-means algorithm for optimization, and did sampling 10000 Boston voters to compare results from each optimization.

1. A k-means algorithm (optByPublicT.py; optByBusstop.py; optByMBTA.py)

   We used a k-means algorithm to find the optimal polling locations based on public transportations, bus stops, and public transit. In each file, we used K-means algorithms to find 255 optimal polling locations in each ward. For optByPublicT.py, we first merged the data sets for bus stops and MBTA and computed a k-means algorithm with both bus stops data set and MBTA T station data set. For the other two files, we computed a k-means algorithm with each data set. Three files return a different list of polling locations in each ward.

2. Statistical Analysis with Sampling and Inference

   Because it is difficult to tell which optimization method is the best without scoring or evaluating locations, we performed statistical analysis with four different lists of polling locations. We randomized 10000 addresses in Boston for voters' addresses, instead of using every voter's address in Boston to compare among polling locations we optimized. By calculating Euclidean distance between randomized voter's address and the nearest polling location, we determined which optimization method provides the highest accessibility to Boston voters. Throughout the distribution of distance between voters and the polling location, scoringLocation.py returns the result of statistical analysis in 95% confidence interval.

Although our optimization improves the accessibility to the polling location, our optimization results are not perfect solution to improve voter turnout. First, a k-means algorithm does not consider the other factors which influence voter turnout. It does not guarantee any usability as polling locations. Moreover, because some polling locations share same coordinates, a k-means algorithm scattered them to separate locations. Therefore, it is hard to determine our optimization results are more accessible than original polling locations.

Visualization

In the scatterplot below (Figure 1), we used blue points to denote “original polling locations” and we used red points to denote “polling locations optimized with public transportation”. Since there are so many points in a single scatterplot, it is difficult for us to compare the difference between these two polling locations and determine which one is better. Same situations also happen when we try to compare the original polling locations, polling locations optimized by MBTA, and polling locations optimized by bus stops. To solve this problem, we decided to visualize on the map separately and compare results in one table.

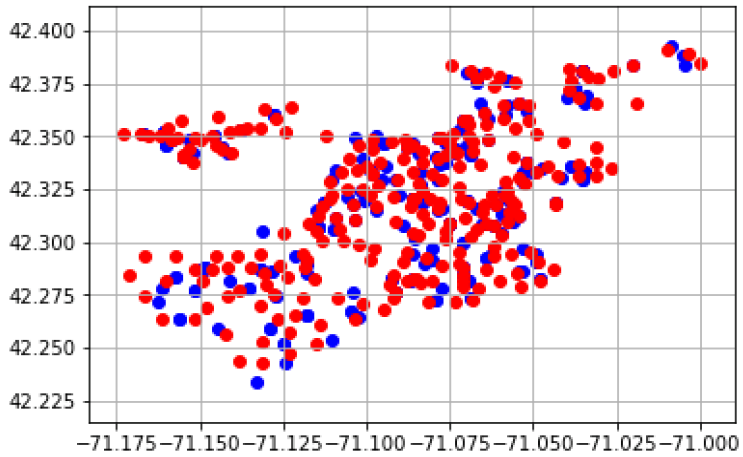

Figure 1: *Scatterplot for original polling locations (blue) and polling locations optimized with public transportations (red).*

In order to visualize our result more clearly, we created an interactive map (Figure 2, 3). In this map, we have two dropdown boxes: one for result, and the other for ward. First dropdown box

has four choices: original polling locations, polling locations optimized by bus stops, polling locations optimized by MBTA, and polling locations optimized by public transportations. For each of these four choices, we can also explore the polling location for every ward or explore one specific ward in more details. For example, we choose “polling locations optimized by bus stops” and “Ward 2” in dropdown boxes, the map will show all the polling locations optimized by bus stops (a k-means algorithm) in ward 2 only. Therefore, users can use this map to filter out all the other polling locations they are not interested in and focus on specific ward.

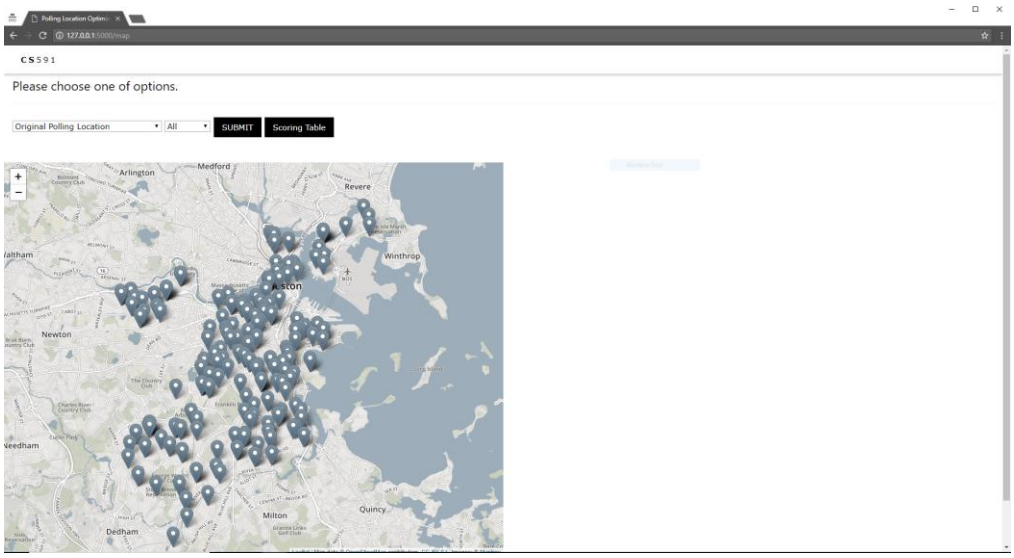

Figure 2: *Interactive map screenshot for original polling locations in every ward.*

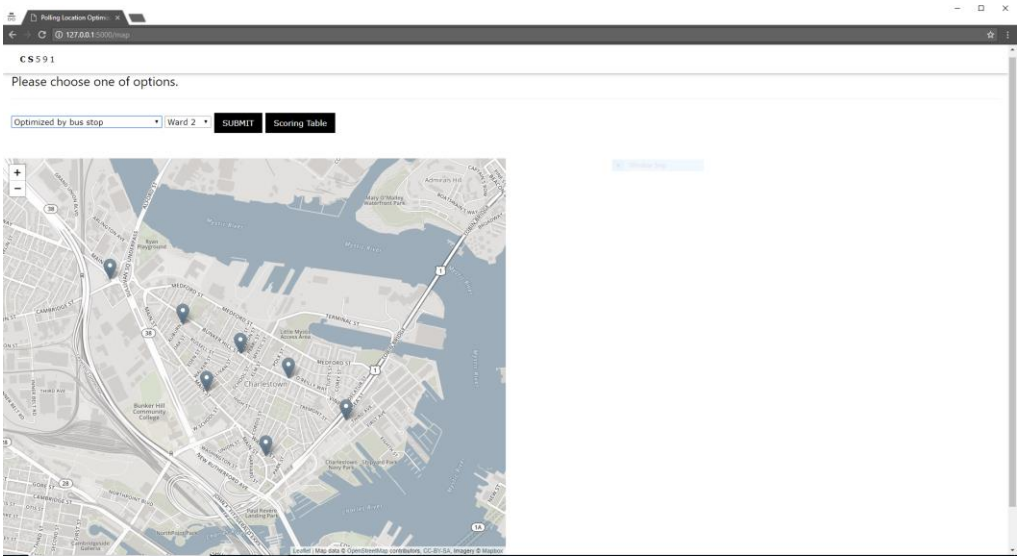

Figure 3: *Interactive map screenshot for optimized polling locations in ward 2.*

In this table below (Table 1), we tried to compare the mean (the distance between the voter locations we randomly generated and the polling locations), the standard deviation, and 95% confidence interval of original polling locations and the polling locations optimized by bus stops, MBTA T station, and public transportations (bus stops and MBTA T station combined). From

the data, we concluded that the original location is enough accessible to the voter locations and there is no big difference between the original polling locations and the optimized ones. However, all three kinds of the optimized polling locations are more accessible than the original polling locations. Moreover, among three kinds of optimized polling locations, optimization with bus stops is better than optimization with MBTA, because the number of bus stops in our data set is much more than the number of MBTA T stations. This can be also explained by the confidence interval. Since the wider the confidence interval, the smaller the sample size, and vice versa. Therefore, the confidence interval for "optimization with MBTA" is wider than that for "optimization with bus stops", because the sample size of the bus stops data sets is much larger. (The width of the confidence interval for "optimization with bus stops" is 2.484, and the width of the confidence interval for "optimization with MBTA" is 2.76.) With this in mind, we can also understand why the data for "optimization with public transportation" is so similar to the data for "optimization with bus stops" because when we merge the bus stop data set and the MBTA T station data set, most portion of the data set are data related to bus stops, but not MBTA T station.

Polling Locations	Average (Miles)	STD (Miles)	95% Lower Tail Confidence Interval (Miles)	95% Upper Tail Confidence Interval (Miles)
Original Polling Locations	1.035	0.828	0.069	2.967
Optimization with Bus Stops	0.828	0.69	0.069	2.553
Optimization with MBTA T station	0.966	0.828	0.069	2.829
Optimization with Public Transportations	0.828	0.69	0.069	2.484

Table 1: *Scoring table to compare original polling locations, polling locations optimized by bus stops, polling locations optimized by MBTA, and polling locations optimized by public transportations.*

Conclusion

As the table (Table 1) shows, each result has an improvement over original polling locations. Polling locations optimized with bus stops only and with public transportations (bus stops and public transit combined) are more accessible than polling locations optimized with MBTA T stations because the number of bus stops is more than MBTA T stations. However, in this research, we found some original polling locations are in the same building but each location is for different people. By running a k-means algorithm we have more markers on the map (Figure 1, 2), which improves the accessible score of each result of optimization. Therefore, we found that the county commission already chose enough accessible polling locations for Boston voters. Still, the work presented here will be developed more for future studies of America's voter turnout problem with considering more factors to increase voter turnout rates.

Future Work

Our next step to improve this research is routing between the nearest polling locations and randomly generated voters' addresses. In this research, we did not care how people would get the designated polling location. Therefore, with considering the route to polling locations, we will be able to determine which polling locations are more accessible to Boston voters. Furthermore, since we ran a k-means algorithm to optimize on public transportations, we did not consider building usage, openness to public, and so on. Therefore, our optimal polling locations might not be ideal polling locations in reality. For example, some polling locations might be too small or

private properties. Therefore, we will develop the scoring methods to evaluate suitability as a polling location.

References

[1] U.S Elections Project. “2016 November General Election Turnout Rates.”, www.electproject.org/2016g
[2] Romano, Rosie. "Increasing Voter Turnout: Can Mass Transit Help?" Clocks and Clouds 1.1 (2012). www.inquiriesjournal.com/a?id=1618
[3] Wallace, Nick. “The Best Cities for Public Transportation.” SmartAsset, SmartAsset, 6 July 2017, smartasset.com/mortgage/best-cities-for-public-transportation