An Optimized Approach to Vaccine Clinic Placement

Connor Douglas[1], Aditya Krishnamachar[1], and Vishesh Patel[1]

Washington University in St. Louis
{c.douglas,akrishnamachar,vppatel}@wustl.edu

Abstract. Determining the optimal placement of stores, factories, and other industrial facilities has long been the topic of much scrutiny throughout the computer science and operations research communities. Indeed, this topic has been studied thoroughly under the name of *the facility location problem*. Yet, when profit is not a driving factor, there is little incentive to scrutinize these placements, which can often involve a costly process of modeling and optimizing over large datasets. As the COVID-19 pandemic rages on, we see the opportunity to leverage past research in this space for the public good.

In this paper, we make two novel contributions. The first is that we put forth and implement a *weighted k-median* algorithm. This algorithm efficiently approximates the position of k clusters in order to minimize the weighted sum of distances to the nearest center for all points in a population. The second contribution is that we apply this model to inform vaccine clinic placement in the greater St. Louis area. Our work is broadly applicable to any population and comes at a time when vaccination rates are insufficiently low and the rate of contagion is high.

Keywords: Facility location problem · Weighted k-median · Vaccination clinic optimization

1 Introduction

The novel coronavirus SARS-CoV-2 - known as COVID-19 - has had sweeping effects across the globe, infecting over 270 million people and resulting in over 5 million deaths, at the time of writing this paper. COVID-19 vaccines have been shown to be safe and effective; guarding against getting the disease as well as developing serious symptoms from it. High levels of vaccination as a society will also protect against both the creation of and sickness from variants like the recently discovered Omicron substrain. As in other contexts - flu, malaria, etc - we know that a decreased distance to vaccination clinics is anecdotally linked to increased vaccination rates. This is evidenced, albeit on a more micro-scale, by the work of Beshears *et al.* Mass vaccination sites in particular have been shown to be useful in this regard (Goralnick *et al.*, 2021).

Vaccinating as many people as possible is undoubtedly a tricky problem, but the same holds with this as for other things out of a person's ordinary day. Voting, going to the doctor's, and getting a car checkup are all similar activities that take time and are in potentially inaccessible or far away locations; hence why some persons do not do not partake regularly or at all. By removing this limitation and having vaccination sites as close by to as many people as possible, we can alleviate this particular concern and continue to increase vaccine uptake.

An optimal placement of clinics, thus, should be our lodestar as we continue to work to get out of the pandemic. This optimal placement is driven mainly by population density statistics. By placing more clinics where there are the highest concentrations of people invariably leads to more vaccinations; but it is not as simple as pointing to dots on a map. Dealing with great distances, different sizes of population centers, and dealing with sparsely populated areas are all problems that we need to address in this work. We take the area of St. Louis City and surrounding regions as our location to focus on; noting both our (the authors) significant connection to the area as well as the low vaccination rates (roughly 50-60 percent) therein. Clearly, there is an opportunity both here and worldwide to achieve meaningful change, and we hope to present a way of doing so.

We theorize that this will lead to increased vaccination uptake, saving lives and improving countless others' physical and economic welfare. We look to solve a model to inform vaccination clinic placement, as the COVID-19 pandemic continues to rage on as well as for any potential future outbreaks of similar diseases. Our model is broadly applicable to vaccination clinic placement as a whole and can be used as an allegory to the greater facility location problem, which has numerous real-world applications.

2 Related work

Previous work in this area centers around two key points – firstly, the relationship between COVID-19 cases and population density; and secondly, previous work regarding so-called k-median algorithms (with median referring specifically to L1-median, though notation differs from source to source). We present this information to provide a clear and holistic picture of the background and rationale behind our work.

We first discuss a detailed study from Brazil, where case counts of SARS-CoV-2 have reached unprecedented highs over a sustained period of time (in the context of this study, the time period ending January 2021). Both anecdotally and empirically, Brazil has struggled to contain the spread of the coronavirus, which unfortunately gives plenty of data to work with in this regard. Martins-Filho *et al.* (2021) present an analysis of the relationship between overall population density and incidence of COVID-19. They first noted that the disease has had an outsize effect in urban and highly dense areas as well as among more vulnerable

populations. Their area of focus was a subsection of northeast Brazil - Sergipe state. The researchers noted that there was a positive correlation both between population density and COVID case incidence as well as mortality rates. The graphs follow here (on a natural log scale). Ganasegeran et al. (2021) found sim-

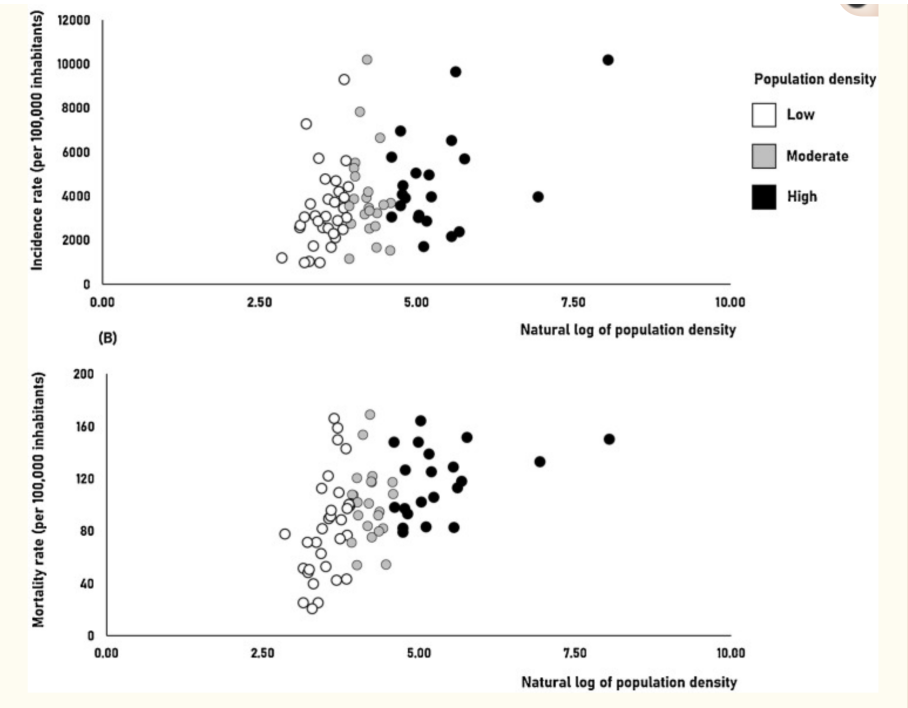

Fig. 1: MDP Portfolio Convergence (Momentum).

ilar results in a large study based on data drawn from Malaysia. More relevant to our specific case can be found in the findings of Smith *et al.* (2021). The latter found that - outside of lockdowns and the subsequent reduction in individuals' mobility - population density along with temperatures were found to be well correlated with increased transmission of coronavirus cases. This analysis utilized comparative regression and 'integrative epidemiological modeling' to come to their conclusions. Their findings included the effects of lockdowns as well, so that the R_0 rate (R_0 is defined as the transmission rate, in this case in the specific context of COVID-19) can be accurately tracked across different periods of time; and is shown to correlate to 'expected' findings. Specifically, that the R_0 rate drops during lockdowns as people spend less time interacting with each other (and by that logic, less time spreading the virus to each other).

The second key area is previous work on k-median algorithms as well as the facility location problem. We drew primarily from papers discussing these topics. Arya, Meyerson, Pandit *et al.* (2001) analyzed search heuristics for k-medial and facility location problems, defining a locality gap as a maximum ratio of (1) some local minimum solution to (2) the global optima. They derive exact values

for this locality gap for different scenarios – local search with facility swapping, and both capacitated and *un*capacitated facility problems (where a location either has finite or infinite capacity). This paper served as an excellent background and grounder for us on the mathematical side of things.

Vardi and Zhang (1999) focused more on deriving the L_1-median of a specific data cloud itself. (We note that the L_1 median is defined to be a point which minimizes the sum of Euclidean distances to any specific point in a given dataset). Their work yielded a monotonically converging algorithm that resulted in the L_1-median; as well as defining depth functions to a greater degree. They derived a modification of the Weiszfeld iterative algorithm - that allows for computation of the multivariate L_1 median. This algorithm serves as an excellent grounder for our specific problem and allowed us to have a solid reference frame for creating our algorithm.

3 Data

3.1 Data Source

For this research, the data source used was the "United States High Resolution Population Density Map" from Data for Good at Meta. The data source consists of six separate .csv spreadsheets named "population_usa.part_n_of_6", where each row of data in each spreadsheet consists of 3 columns that provide information on latitude, longitude, and population respectively. Each row corresponds to a point in the United States that is representative of a 30x30 meter grid and provides population density information on that grid. Each spreadsheet consists of over 31 million data points, and in total contains 191 million data points with population density information. According to the owners, the data is sourced using machine learning image processing technology that scans commercially available satellite imagery to identify buildings. Once those locations are flagged, population estimates are then generated by researchers at Columbia University based on available population statistics and US census data.

3.2 Refining Geographical Search Space

The initial research plan was to investigate what the most optimal locations to place COVID vaccination clinics were across the entirety of the continental United States based on population densities. However, given the volume of data points, the area of interest was reduced to the state of Missouri. When reducing the search space to Missouri, we developed a bounding box that fully encompassed all boundaries of the state. When filtering the dataset based on the bounding latitudinal and longitudinal points, the dataset was reduced to 9 million data points. Furthermore, the bounding box was rectangular in shape and was formed using the most extreme points in Missouri's state boundary, not in terms of the shape of the state itself which resulted in an inclusion of

points in other neighboring states like Illinois and Kansas. We then narrowed our search space down to the greater metropolitan area of St. Louis. We manually determined a bounding region and determined the bounding geographical coordinates. After filtering the six datasets, the dataset was left with about 670,000 points. We further removed the 70,000 data points with population densities weighted at zero resulting in 600,000 data points. Data cleaning and processing was done in Python3 using the Pandas data analysis library. Each spreadsheet was downloaded and iteratively filtered. Using the bounding latitude and longitude coordinate values, each dataset was filtered down to only consist of points within the specified ranges. Each subsequent data frame was then merged to create a singular dataset that contained all the data points for the areas within the boundary we specified around the Greater St. Louis Metropolitan Area. This same approach was applied using coordinates that bounded the state of Missouri.

3.3 Coordinate Conversion Methodology

In order to run our algorithm, the bounding box we placed around St. Louis needed its coordinates converted into a standardized metric. While viewing an image of St. Louis on a map may look two-dimensional, the nature of the Earth's shape results in a difference in units based on the latitudinal and longitudinal axes. A methodology of approximating the distance between two points on the globe is given by the Haversine equation. We applied the Haversine formula on the coordinates that bound the St. Louis region to determine the distance in kilometers between each of the four points that bound the region. Next, we set the bottom-left point of the region to (0,0) and converted each coordinate to a point that falls within the boundary in kilometers. We approximated the longitudinal distance to be 57.866 kilometers. In a larger region, however, the horizontal edge of the boundary that is closer to the equator will have a larger width than the horizontal edge further away from the equator.

4 Model

4.1 Objective Function

At the core of our model, we try to minimize our objective function. This function is the weighted Euclidean distance of n points in a dataset to the nearest k cluster center. We refer to the set of points in the population as P and the set of clusters as C, where each point is a tuple consisting of a x and a y value. Each point in the population is assigned some weight, w_j.We treat the points in the population as fixed, whereas the points in C are variable. We refer to the set of points which have center i as their nearest center as P_i where $P_i \subseteq P$. From these notations, we arrive at our formal definition of our objective function, Weighted Distances To Nearest Cluster (WDTNC), below.

$$WDTNC(C) = \sum_{i \in C} \sum_{j \in P_i} w_j \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{1}$$

To minimize this function, we search along x and y values for all k points in C. At first, this function seems relatively easy to minimize. We search along these 2 dimensions, x and y for each of k points, making for a total search space in $\mathbb{R}^{2k}$. Moreover, these values are bound to the range of values in P, narrowing the search space. Unfortunately, perturbations to values in C change cluster assignments for points in P making this objective function discontinuous. Because of this, gradient-based approaches are infeasible, where computing an exact solution becomes intractable as k grows. To solve this, then, we turn to an iterative approach.

4.2 The Weighted k-Medians Algorithm

The *weighted k-medians* algorithm looks to minimize the objective function laid out above, WDTNC. It is partly based on the *weighted k-means* algorithm, but where this algorithm uses the weighted centroid of a population to assign clusters, we use a weighted geometric median. The former is much more trivially calculated than the latter, as the weighted centroid is simply the mean value across all dimensions of the weighted population. At a slightly more intuitive level, it is worth noting that the centroid minimizes the sum of squared distances from a center to all points, while the geometric median minimizes the Euclidean distance, which is of greatest concern to our application in vaccine clinic placement.

We also build off of the *k-medians* algorithm, a lesser-known cousin to *k-means*. The key difference here is, as laid out before, the use of a geometric median as opposed to a centroid of a set of points. Yet, every point in our data does not represent one individual in the population. As discussed in section 3, each point is given a population density value, corresponding to the number of residents in a given 30 meter by 30 meter tile. It is necessary to include this weight as not doing so would wildly skew our results towards rural areas. Moreover, if we hope to incorporate further weighting based on some function of population density, as laid out in section 2, an algorithm that includes weights is even more pertinent.

Surprisingly, to our knowledge, no *weighted k-median* algorithm exists publicly. Because of this, we sought to create our own algorithm in order to minimize our objective function. In words, this procedure follows an Expectation-Maximization approach. We iteratively fix the assignments of points in a population to the nearest center. Then, we recompute centers based on the weighted geometric median of the subset of the population assigned to each cluster. This process converges once assignments do not change between successive iterations. The pseudo-code for this algorithm is laid out below. Here, w refers to the list of weights for each point in the population and k is the number of centers. Note, X refers to the fixed population, instead of P as used above, for consistency with programming conventions.

The getAssignments() Function The formulation of this function is relatively straightforward. For this function, we calculate the distance matrix from

Algorithm 1 Weighted k-Medians

```
1: procedure getWeightedKMedians(X, k, w)
2:     medians ← randomChoices(X)
3:     assignments ← getAssignments(X,medians)
4:     while ! convergence do
5:         assignmentsOld ← assignments
6:         assignments ← getAssignments(X,medians)
7:         medians ← getResults(data,weights)
8:         if assigmentsOld = assignments then
9:             convergence ← true
10:        else
11:            oldResults ← results
12:        end if
13:    end while
14:    return medians
15: end procedure
```

points in *medians* to points in X. This leaves us with a n x k matrix of distances, where each row corresponds to a point in the population and each column corresponds to a given center. The value of each cell is the Euclidean distance from a given point in the population to a given center. From this, we take the *argmin* for all rows to ultimately return a list of length n, where each value is in the domain $[1, k]$, corresponding to the point's assignment.

The getWeights() Function This function leverages recent work done by Vardi and Zhang to quickly compute the weighted geometric median of a set of points. As this formulation is fairly technical, we have omitted the formula for the sake of brevity. We compute this point for each center, i, in $1...k$ subsets of the population, with each subset being the points in the population nearest to center i.

4.3 Weighted k-Medians for Vaccine Clinic Placement

We now tie our algorithm formulation back into the context of our original problem. We use the *weighted k-medians* algorithm to minimize our objective function, WDTNC. In a simple model, we still have to assign weights to each point, as a point is representative of a tile of people, with potentially multiple people living in each tile. By assigning a simple weight of *population density*, we capture this and create a model that minimizes the distance to all people in the population, not just all points in the dataset.

In our more sophisticated model to mitigate virus spread, we choose weights via a different method. To select our weights, we tie in domain knowledge of how epidemics spread. From section 2, we see that contagion rate is proportional to population density. So, in order to minimize contagion rate by lowering barriers to vaccination, we want to assign extra weight to higher density areas. For this we choose $\textit{population density}^2$ to be the weight for each tile.

5 Results and Evaluation

In the evaluation of our model, we plot against the greater St. Louis area population. In lieu of having a strong alternative to act as a benchmark for our model, we demonstrate visually how our model performs. To begin our analysis, we look at an elbow plot of the average distance to the nearest clinic against the number of clinics placed. From this, we surmise the most cost-effective number of clinics. This elbow plot is show in Figure 2. From this figure, we see that the average

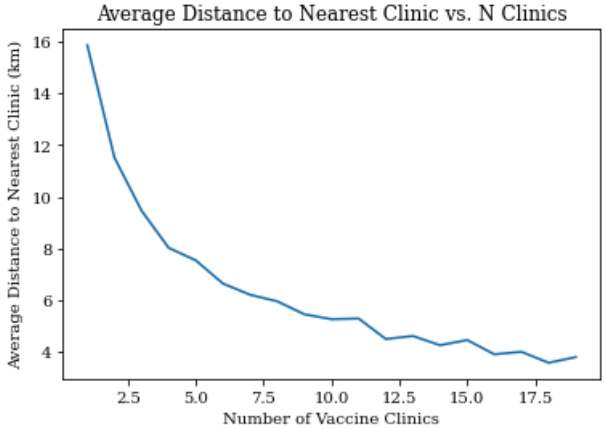

Fig. 2: Elbow plot of distance to nearest center for k center numbers

distance to the nearest clinic starts to flatline at a k value of roughly 25. Assuming that there is a constant cost to creating vaccination clinics we choose this value as it promises to roughly minimize distance per spend on clinic creation. The optimal k value could be subject to more scrutiny in the future, but for the scope of this model, we suffice with these grounds. In Figure 3 we see the optimal solution from our *weighted k-medians* weighted only by population density. In this and subsequent figures, red dots are points in the dataset, with color intensity corresponding to population density (more red implies higher density). Blue dots correspond to clinic locations determined by our algorithm. This results in an average distance to the nearest center of $2.68km$ across the whole population of just under 1.5 million people. Again, this weighting effectively makes our solver treat this problem as an unweighted optimization to all individuals in the population. But, because our data is clustered into tiles and given a population density per tile, we incorporate this density as an initial weight.

We then contrast this figure with Figure 4 which further weights each point by another factor of population density, to make the weighting *population density*^{2}. This slightly biases centers towards higher population areas. While the differences are only slight, we see fewer centers placed in the East St. Louis area (roughly $x > 40$), which is known to be less dense than locations west of the Mississippi River (the white stripe undulating vertically at around $x = 40$). The St. Charles area (top left, north of the Missouri River), looks nearly identical between graphs, while the downtown St. Louis area (high density area in the center) receives a couple more clinics.

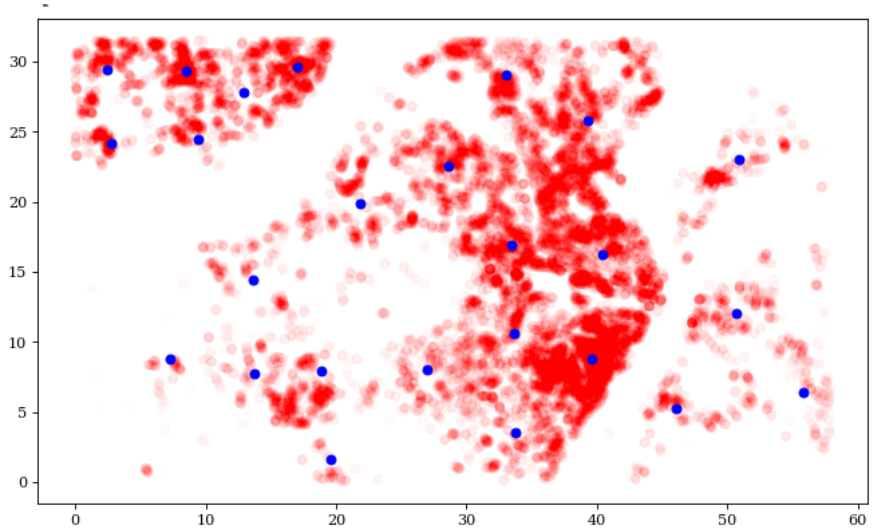

Fig. 3: k-medians weighted by population, $k = 25$

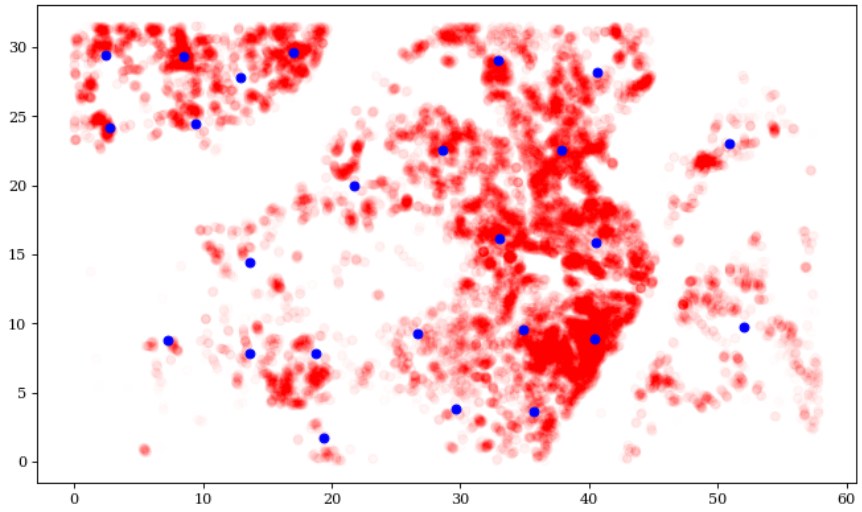

Fig. 4: k-medians weighted by *population density*^{2}, $k = 25$

We then scale up our weighted model to $k = 100$. This is more consistent with the magnitude of clinics currently in the region, which, according to KMOV4, sits at around 350. Because 350 clinics would be cumbersome to visualize, we prune this down to an even 100 clinics. In Figure 5, we see that it does accurately cluster around dense areas. which is promising to demonstrate a realistic scale of optimal clinic placement.

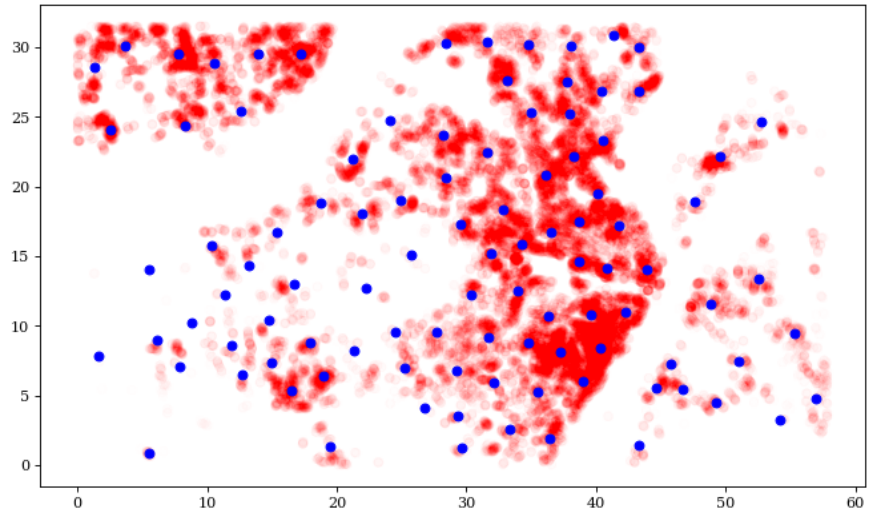

Fig. 5: k-medians weighted by *population density*^{2}, $k = 100$

6 Discussion

Through this paper we successfully optimize vaccination clinic placements in the greater St. Louis region. We first propose a model for a *weighted k-medians* algorithm to solve the challenging problem of minimizing distances for weighted points in a population to a nearest center. We then apply this model to vaccine clinic placement, selecting weights informed by domain knowledge of how viruses spread. For data, we refine our dataset to the St. Louis area, and then transform it from geographic coordinates to a metric-unit approximation. We then combine these into an encompassing model for optimizing vaccination clinic placement. While our research was specific to COVID vaccination clinics, its applications can be extended to other problems that involve optimally placing infrastructure in areas based on regional population density and other kinds of constraints.

Recommendations for Future Work Our research is adaptable and can be expanded upon in a few different ways. Firstly, from an implementation standpoint, work can be done to optimize distances given monetary budget constraints instead of a k value. This could incorporate geographic variations in costs to implement clinics. Secondly, further research may consider constraints on this optimization, such as vaccination rate caps or total supply caps. Additionally, subsequent projects should analyze this problem using different solvers. While we developed our own solver, there may be other solvers and algorithms that could be applied to this objective function. Further research would use these solvers and compare the results to those of our own.

Finally, as stated above, our methodology can easily be applied to other areas in the United States, be they cities, towns, counties, or even states. Further research in other regions can also use a scientific approach to determining the

appropriate bounding coordinates and regions for the area of interest. This also pertains to weighting. While our model assumes a somewhat naive relation between population density and contagion rate, this assumption may break down on the micro level due to the high mobility of populations within the search space. Epidemiological research on the true contagion rate at different geographic scales could help inform this weighting system.

7 Division of labor

Connor Douglas Connor was responsible for the algorithm design and implementation along with empirical testing. His sections were *Model* and *Results and Evaluation*.

Aditya Krishnamachar Aditya dealt with much of the background reading on this subject and the motivation for certain approaches we take. Aditya's primary sections were *Introduction*, *Related Work*, and *References*.

Vishesh Patel Vishesh was the team's data engineer. He found our dataset and was in charge of verifying its accuracy, refining the geographic range of data, and transforming the data. In writing, he worked on *Data* and *Discussion*.

8 References

Aloupis, Greg. Robust Estimators of Location — The L1 Median, http://cgm.cs.mcgill.ca/ athens/Geometric-Estimators/L1med.html.

Beshears, John PhD*; Choi, James J. PhD†; Laibson, David I. PhD‡; Madrian, Brigitte C. PhD§; Reynolds, Gwendolyn I. MTS Vaccination Rates are Associated With Functional Proximity But Not Base Proximity of Vaccination Clinics, Medical Care: June 2016 - Volume 54 - Issue 6 - p 578-583 doi: 10.1097

"Covid-19 Vaccine: Here's Where to Find It in the St. Louis Area and How to Sign Up." KMOV.com, KMOV, 9 Sept. 2021, https://www.kmov.com/news/covid-19-vaccine-heres-where-to-find-it-in-the-st-louis-area-and-how/article_d9b094fe-5784-11eb-a0f1-f315004a17c7.html.

Ganasegeran, Kurubaran et al. "Influence of Population Density for COVID-19 Spread in Malaysia: An Ecological Study." International journal of environmental research and public health vol. 18,18 9866. 18 Sep. 2021, doi:10.3390/ijerph18189866

Goralnick, Eric, et al. "Mass-Vaccination Sites - an Essential Innovation to Curb the COVID-19 Pandemic: Nejm." New England Journal of Medicine, 2021, https://www.nejm.org/doi/full/10.1056/NEJMp2102535.

Hu, Hao, et al. “The Scaling of Contact Rates with Population Density for the Infectious Disease Models.” Mathematical Biosciences, Elsevier, 9 May 2013, https://www.sciencedirect.com/science/article/pii/S0025556413001235.

Martins-Filho, Paulo R. “Relationship between population density and COVID-19 incidence and mortality estimates: A county-level analysis.” Journal of infection and public health vol. 14,8 (2021): 1087-1088. doi:10.1016/j.jiph.2021.06.018

Smith, Thomas P., et al. “Temperature and Population Density Influence SARS-COV-2 Transmission in the Absence of Nonpharmaceutical Interventions.” PNAS, National Academy of Sciences, 22 June 2021, https://www.pnas.org/content/118/25/e2019284118.

Vardi, Y, and C H Zhang. “The multivariate L1-median and associated data depth.” Proceedings of the National Academy of Sciences of the United States of America vol. 97,4 (2000): 1423-6. doi:10.1073/pnas.97.4.1423