## 

# 

OPTIMIZING FOR COVID-19 VACCINE DISTRIBUTION CENTERS IN 2021

## 21-393 Operations Research II Fall 2022

## Sai, Andrew, Lisa, Eli, Rohit, Matthew

## 

## 

## 

## **TABLE OF CONTENTS**

[**ABSTRACT**](#_813kl7umodu5) **3**

[**INTRODUCTION**](#_bd67phmsk6e6) **3**

[**METHODOLOGY**](#_dfm5kdm607l6) **3**

[Data Collection / Formatting](#_3jbwkt1zbtne) 4

[Weighted K-Means](#_c03bo46ye16h) 4

[Dynamic Programming/ Inventory Programming](#_e2r9rkfpymh9) 5

[**DISCUSSION**](#_bmojpxoelsgc) **6**

[Weighted K-Means](#_4thz3lukyull) 6

[Dynamic Programming / Inventory Planning](#_pmphp0qoy94t) 8

[**CONCLUSION**](#_n8ouxbdmxv01) **9**

[Limitations](#_j4gi7qwrb2st) 9

[**REFERENCES**](#_qayudyjkp3ip) **11**

## 

## **ABSTRACT**

The purpose of this project was to use mathematical models to optimize COVID vaccination distribution for distance and cost efficiency across the United States. To investigate this, we used two different methods: weighted k-means clustering and inventory planning. The k-means clustering gave results of 7 distribution center locations using an SSE minimization algorithm. We then utilized these locations along with other factors in inventory planning to come up with the minimum cost for each possible circumstance of the initial number of stored vaccines.

## **INTRODUCTION**

Since the beginning of 2020, the COVID-19 pandemic has been the root of a myriad of widespread societal issues. With the proliferation of the spread of the virus, and as vaccines began to be developed, tested, and released to the public, it became a top priority to effectively distribute these vaccines. It is now imperative that vaccines are able to be distributed across the country in a manner that is both efficient and cost effective in order to effectively combat the spread of COVID. **Our aim is to use mathematical models to optimize COVID vaccination distribution for distance and cost efficiency across the United States.**

In our investigation, we plan to calculate distribution center numbers and locations across the contiguous United States and create an inventory plan to optimally distribute vaccine shipments to each individual county. We will simulate the COVID situation from January 1st, 2021 to December 31st, 2021 by utilizing historical COVID data in 2021 from reputable sources.

## **METHODOLOGY**

To investigate our problem, we adopted two main methods of optimization for COVID vaccination deliveries:

1. A k-means algorithm that optimizes for the number of distribution centers and distribution center locations, and
2. A Dynamic Programming algorithm that calculates inventory planning with the known inventory clusters concluded from the k-means algorithm, maximum capacity of vaccine batches, and cost function.

We mainly utilized Python and R programming languages and packages to engineer our algorithms.

### Data Collection / Formatting

To collect and use our data in our algorithms, we utilized R to convert our data into data frames for accessible manipulation. We used data from three sources to combine county midpoint coordinates, COVID cases in individual counties per month, and vaccination rates in individual counties per month.

Here, we note that some COVID data in certain counties were not available in the datasets. As a result, data from Kansas, Kentucky, and Louisiana are absent. Additionally, we have restricted our dataset to only focus on the contiguous United States, excluding Alaska, Hawai’i, and Puerto Rico, so that our centering algorithms will not be affected by outliers.

### Weighted K-Means

To optimize vaccine distribution center locations, we decided to use a weighted k-means algorithm. To optimize the number of locations, we utilized an elbow method analysis of the average sum-of-squares error (SSE) of the results.

A k-means algorithm will allow us to calculate centroids of clusters according to the locations of each county. Here, we identify and approximate county locations with the longitude and latitude coordinates of their midpoints. We will use these coordinates in an x-y plane for our algorithm.

However, we note that consideration of each county will not yield effective results, as we observe that some counties report more cases than others. Therefore, we will use a modified, weighted k-means algorithm, with the weights being total COVID cases per county in 2021.

Before we run our algorithm, we define 2 important functions that are necessary to accurately run our weighted k-means algorithm:

1. Haversine Distance Function: calculates the distance between two points on the surface of a sphere. This takes longitude and latitude measurements into consideration and yields a more accurate estimate of distance given the curved nature of the Earth.
2. SSE + Elbow Function: To determine the optimal number of distribution centers, *k*, we will utilize the elbow method in clustering. We calculated the SSE for values of *k* from 1 to 10, which in this case refers to the deviation between a data point and its assigned centroid. We calculated this error 10 times for each value of *k* and then averaged all of the SSEs for an individual value. This then indicated a level of accuracy for each value of *k*, as well as controlled for the randomness of the k-means algorithm and the potential pit-fall of finding local minima instead of a global maximum. The optimal value of *k* was then chosen, not by which average SSE was smallest, but by where the decrease in SSE between values of *k* started to be less significant.

Our k-means algorithm follows this general form:

1. Initialize *k* clusters in random locations.
2. Compute distances between all data points and the existing *k* centroids and re-assign each data point to the nearest centroid accordingly.
3. Calculate the new *k* centroids by taking the mean of the data points assigned to each cluster.
4. Repeat steps 2 and 3 until the newly calculated centroids converge onto the same point or until max number of iterations.

Our algorithm returned a list of seven coordinates for each cluster, as well as a cluster assignment (1 through 7) for each data point in our original dataset.

### Dynamic Programming/ Inventory Programming

To determine the minimum cost of vaccine deliveries in terms of time and quantities, we used inventory planning based on the results from k-means clustering. Our goal was to use the demand, generated cost function, and previously calculated distribution center locations to come up with the optimized production solution. In order to narrow down our data set to what was feasible for us to work with, we focused on inventory planning for the year of 2021, using each month as a time period and the number of vaccines administered in that time as a measure of demand.

It was necessary for us to complete inventory planning for each facility chosen by our k-means algorithm, so the first step in our data processing required us to separate the demand date by cluster. After doing so, demand was aggregated by month. This essentially provided us with a “demand function” by giving a table for us to pull a monthly demand from.

We also had to develop a cost function. Our function took the following form:

,

where is the maximum capacity of batches, , and is the fixed costs per week that include specialized storage, equipment, staff time, carrying costs, etc. The assumption we make here is that if we need batches, filling the entire capacity maximizes the cost and then starts decreasing afterwards. In other words, if we produce twice the capacity, then it would be equivalent in cost to make nothing as we make nothing the following week.

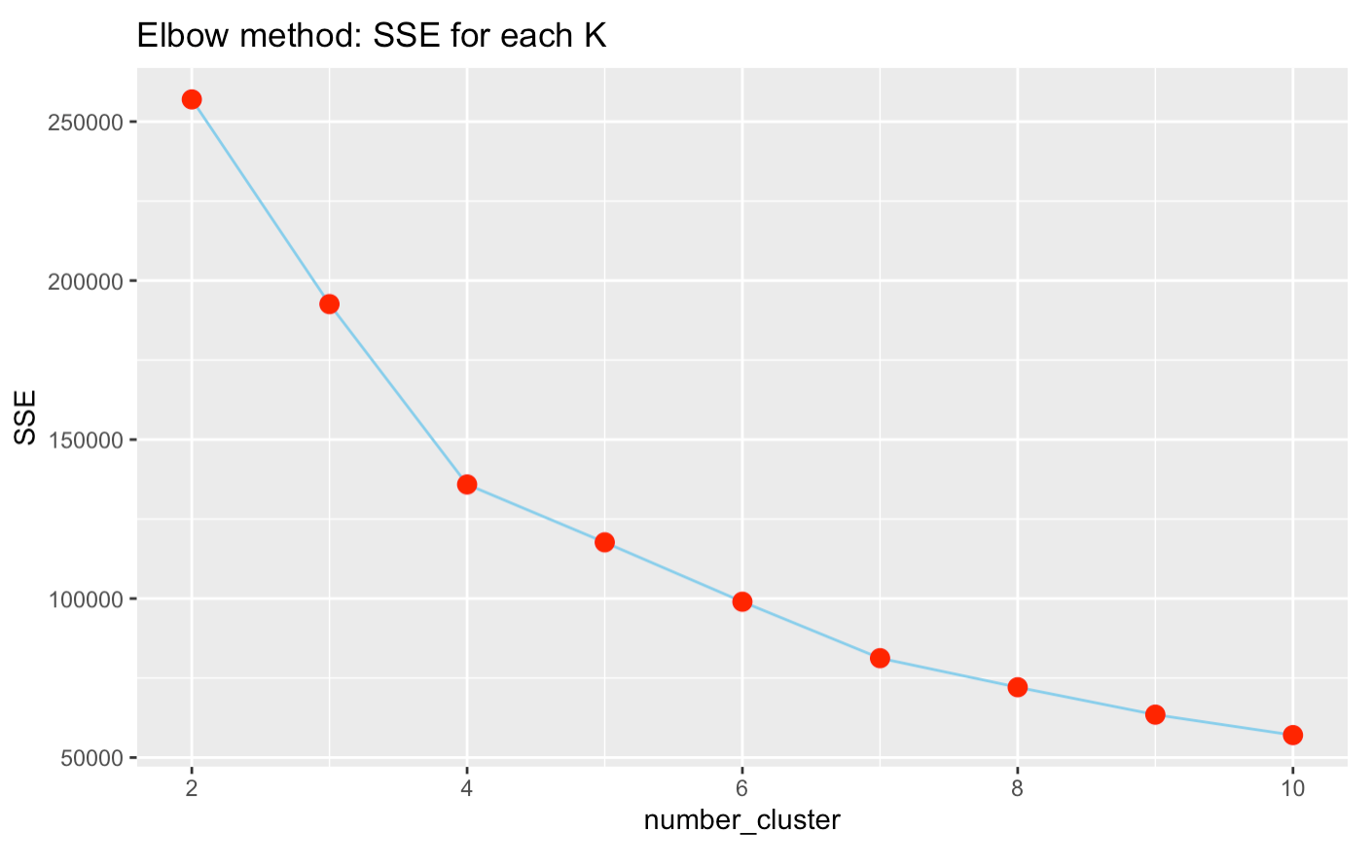
With these defined functions, we were able to begin backward induction. This assumes that in the final period, we are interested in producing the necessary amount of vaccines to meet demand without holding onto any remaining inventory. We defined , for each , representing the minimum cost needed to meet the demand of the inventory with the current batches. For t = 12, we then had that , 0), where x is the total number of vaccines required for that time period and b was the batch size we selected, which was 7114 vaccines. If we know , then for any , we have the relationship between and to be:

At any inventory level, we must produce at least the amount demanded by that time period but not covered by the inventory. If we produce more than that amount, the remainder is stored in inventory and we try to minimize cost across all these possibilities.

## **DISCUSSION**

### Weighted K-Means

After running our SSE elbow function, we determined that a value of optimized the number of distribution centers. As can be seen in Figure 1, though the average SSE for is not the smallest out of all of the values of *k*, it is where the decrease in average SSE starts to drop off.



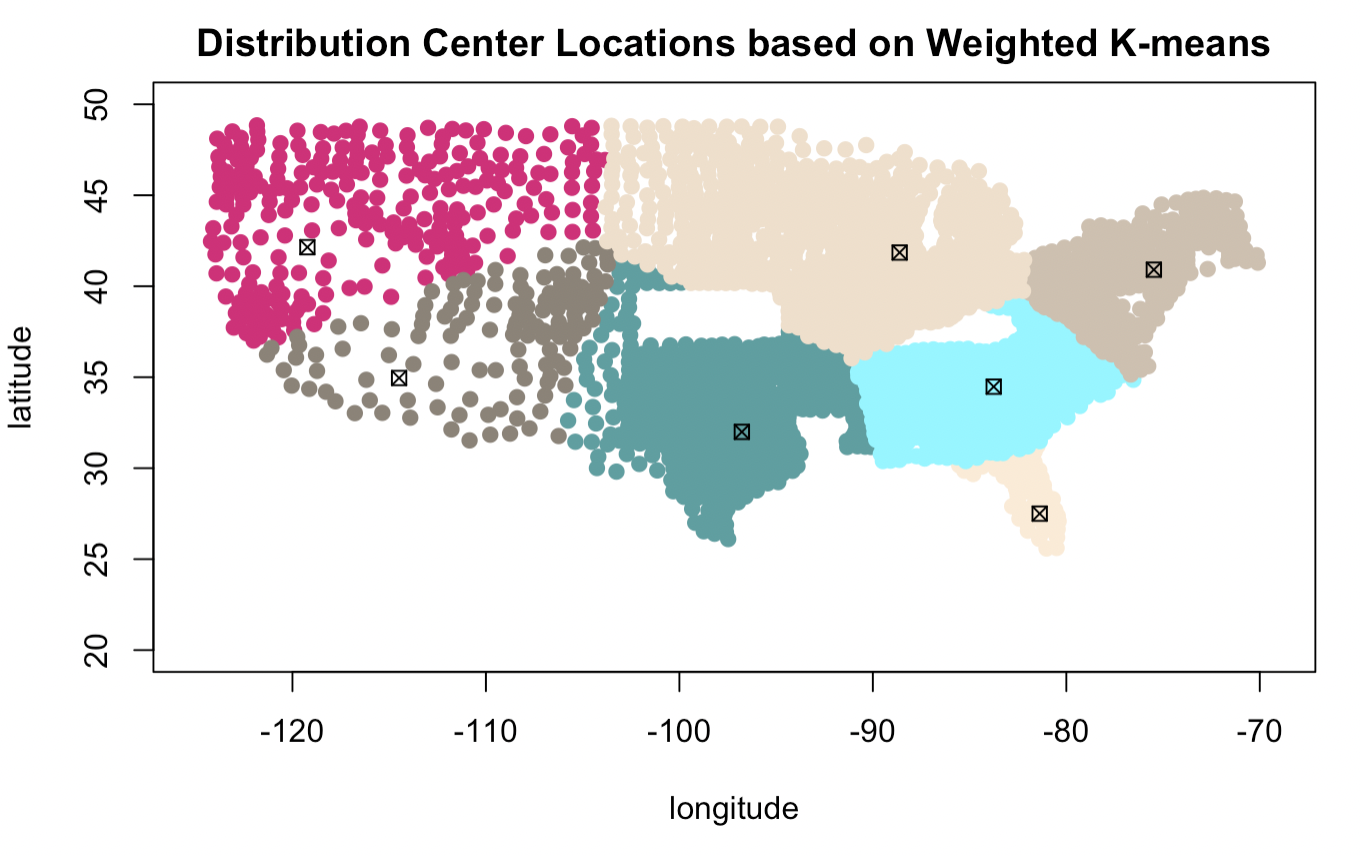
**Figure 1**: SSE for each number of distribution centers

In deciding that 7 locations was ideal, we wanted to think about not just the average SSE, but also about how long the code would run and the potential costs of running so many facilities. We don’t see a significant improvement in optimality by using more than 7 distribution center locations for the costs we incur in computing efficiency and the actual operating costs of the facilities. That is why, even though 10 locations gave a smaller average error, we went with less facilities.

After running our weighted k-means algorithm with our value of *k* optimized, we produced the seven optimal distribution center locations. These are indicated by the boxes with an “x” in Figure 3. The seven facilities were found to be located in Blaine County, ID, Kane County, IL, Hill County, TX, San Bernardino County, CA, Highlands County, FL, Monroe County, PA, and Hall County, GA. The latitude and longitude coordinates corresponding to these locations can be seen in Figure 2. These results optimize for location as well as COVID demand, as it takes into account the total error in a weighted function for all counties to the cluster.

| Cluster | Latitude | Longitude | County | State |
| --- | --- | --- | --- | --- |
| 1 | 43.29°N | 113.63°W | Blaine County | Idaho |
| 2 | 41.78°N | 88.42°W | Kane County | Illinois |
| 3 | 32.06°N | 97.31°W | Hill County | Texas |
| 4 | 34.95°N | 117.31°W | San Bernardino County | California |
| 5 | 27.50°N | 81.38°W | Highlands County | Florida |
| 6 | 40.91°N | 75.48°W | Monroe County | Pennsylvania |
| 7 | 34.46°N | 83.78°W | Hall County | Georgia |

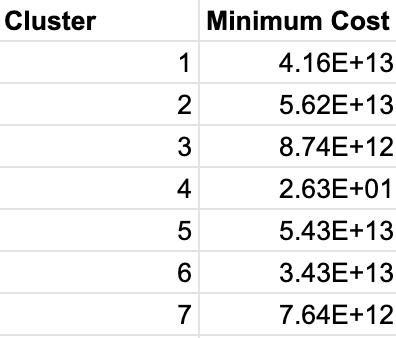
**Figure 2**: Table of the 7 distribution center locations



**Figure 3**: Map of distribution centers and their corresponding clusters

### Dynamic Programming / Inventory Planning

After running the inventory planning, we have a 2-dimensional array that contains all , with being specific month and being the number of possible demand vaccine batches. What we need to look into is the first row here, , which gives the minimum cost for all the possible initial storage circumstances at the beginning of the year. Figure 4 shows this minimum cost starting with an empty inventory at the beginning of January for each distribution center.



**Figure 4**: Required cost (in dollars) of fulfilling the vaccine demand of each distribution center

## **CONCLUSION**

If we look at the results of our k-means algorithm, while in theory these should be our optimal cluster centers, when looking at the coordinates generated on a map, we realized that many of these locations were infeasible. For example, the cluster center in Idaho is located inside of a National Preserve and the center in Illinois is located in the middle of a residential area. Although it seems practical to create a specific radius around the optimal coordinates and choose a feasible location within that region, mathematically this could result in potentially not selecting the optimal feasible location for the facilities. However, due to the lack of complete COVID data for some counties and even entire states, our model only provides us with an approximate solution thus making it a reasonable assumption to choose a feasible facility location within a given distance from the centroid.

Looking at the visual representation of our solution, it seems inefficient to have facilities in neighboring states, as clusters 5 and 7 have their cluster centers in Florida and Georgia respectively. This could potentially be caused by a number of factors, including a high COVID case rate in Florida coupled with the large population of the state skewing the data towards that region. On a macro level, we see that our optimal center locations are concentrated towards the coasts and states with large populations, as six of the seven cluster centers are located in one of the eight most populous states. Thus, our generated optimal distribution facility locations appear to be reasonable, considering the population and COVID data we utilized.

Our inventory planning results, on the other hand, do not appear to be as feasible, as all of the minimum costs required to satisfy demand for each cluster are in the hundreds of billions to trillions of dollars range, except for cluster 4. This indicates that the assumptions we made for our fixed costs when calculating total inventory cost were too high. However, our results are still valuable as the costs provide us useful information about the relative demand of each cluster and relative inventory cost required to satisfy that demand.

### Limitations and Future Research

Two related roadblocks we ran into while working on this project were the lack of data from a few regions and the influence of the non-continental states and territories on the results of warehouse locations. The data we found on COVID cases did not contain any information from the states of Kansas, Kentucky, and Louisiana. We did not find separate data sets to fill this hole, so the clustering results were biased from a lack of data. In future research, data should be found to fill these holes.

Similarly, the results were not accurate in accounting for Puerto Rico, Alaska, and Hawai’i. Including these places would have significantly influenced the final warehouse locations, pulling the centers closer to these remote areas. This is why we removed data for them. However, this means that our final results are not as helpful in a real context because, in actuality, we do care about these places receiving vaccines, and the warehouse locations our code spat out would not be optimal for them. If these remote locations were included in future research, potentially using a larger value of *k* could help with the outlier problem.

One strong limitation we faced with inventory planning was defining the time periods with which we could apply the dynamic program. We had to limit our data set to a feasible amount of time for which we could choose standard time periods. This led to us choosing the full data set for the 2021 calendar year and defining the intervals to be monthly Additionally, the motivation behind backwards induction relies on assuming that, in the final time period, no vaccines will be held in inventory. Realistically, vaccine production would not stop in December of 2021 due to continued demand, but this restriction was necessary for us to set an end point to the algorithm.

We additionally faced a runtime issue while doing inventory planning. For such a problem, an sized matrix is initialized to hold the problem data. Here, for all of the months which we were planning for. The other variable, *I*, is defined as the maximum inventory which any facility is capable of holding. Initially, we wanted to define this value as nearly one million as an attempt to match what facilities can realistically hold with certain fixed costs. We found that this made running the algorithm impossible, so we instead took the approach of defining batches, using 10,000 as the maximum value of possible batches. We found this was a value for which our algorithm terminated in a reasonable amount of time. In order to define how many vaccines were produced per batch, we found what minimum value was necessary to meet the highest demand across all clusters assuming that there is no stored inventory at month 12. These were all major restraints which we had to set, but allowed us to ultimately run the algorithm.

## **REFERENCES**

* https://raw.githubusercontent.com/nytimes/covid-19-data/master/us-counties.csv
* https://www.census.gov/geographies/reference-files/time-series/geo/gazetteer-files.html
* https://data.cdc.gov/Vaccinations/COVID-19-Vaccinations-in-the-United-States-County/8xkx-amqh/data
* https://towardsdatascience.com/using-weighted-k-means-clustering-to-determine-distribution-centres-locations-2567646fc31d
* https://trumedsystems.com/blog/true-vaccine-costs-for-practices/