ME308 Course Project
Optimization of Vaccination Centres: A Set Covering Problem Approach

Soham Purohit, Shantanu Shukla, Boda Sharath Pavan, Anuradha Meena

October 10, 2022

Abstract

With the recent occurrence of an epidemic and a rise in demand for vaccination, there inevitably arises a need for efficiently managing the supply of vaccines, along with the cost involved. In this report, we aim to optimize the location of vaccination centres in a basic model of a city, such that the entire city gets covered with the minimum number of centres. We model this problem as a set covering problem and look to solve this problem with two methods and compare them - a heuristic algorithm and using the CPLEX solver with an AMPL formulation.

1 Introduction

Several waves of COVID-19 have shown us that there can be instances of severe shortage of vaccines at times. Further, there are large costs associated with storage, transport and wastage of the vaccines, which cannot be managed during times of crisis. This fact is amplified for a resource constrained country like India, where vaccine shortage or wastage cannot be treated lightly. With the goal of minimizing these costs, we decided to optimize the location of vaccination centres, on the basis of several assumptions. We modified or relaxed these assumptions to make our model as close to reality as possible.

We base our project on the set covering problem, which is: Given a set with a collection of elements and subsets of this set, we aim to find the minimum number of these subsets that incorporate (cover) all of these elements. Section 2 introduces the base problem. Section 3 presents the mathematical formulation for both the base case and the modified case. Section 4 presents an implementation of a greedy algorithm as an alternative to finding the optimal solution. Section 5 presents the results of the implementations, with some inferences. Finally, 6 presents a sensitivity analysis of the solutions.

2 Problem Statement

Consider a city modelled as a 10x10 grid with a total of 100 cells. There is population present in each cell. We consider that hospitals are randomly distributed throughout this city with a 40% probability of there being a hospital in each cell. Each hospital has a "radius" that it can cover, effectively, a fixed set of cells around the hospital. The population in these cells are catered to by the hospital. For this project we consider a 5x5 grid around each hospital, with the hospital at the centre, as the set of cells each hospital can cover. At a preliminary level, we consider that only hospitals can be activated as vaccination centres, with no restriction on the capacity of each hospital as a vaccination centre. In this scenario we essentially aim to choose appropriate hospitals from the given ones to be activated as vaccination centres so that we are able to cover the entire 10x10 region with the minimum number of hospitals. A visual representation of the problem is provided in Fig. 1.

0	0	0	0	0	0	0	0	1	0
2	0	3	0	4	5	0	0	0	6
0	0	0	0	0	0	0	0	0	7
0	0	0	0	0	0	8	0	0	9
0	0	0	0	10	0	11	12	0	0
0	0	13	14	15	0	0	0	0	0
0	0	16	0	0	0	17	18	19	0
20	0	0	21	0	0	22	23	0	24
25	0	0	26	0	0	0	0	0	27
28	29	0	30	31	32	0	33	0	0

Figure 1: The numbered cells 1 to 33 are the cells with a hospital present. Cells with a 0 assigned have population present but no hospital present. The dark green cell no. 10 is the cell with the hospital we are referring to whereas the 5x5 light green cells around it are the cells hospital 10 can cater to.

3 Mathematical Formulation

3.1 Base Case

In this case, there is no maximum capacity associated with any hospital. Hence the main parameter that dictates the optimal locations of vaccination centres would be the overlap of the area of coverage of the hospitals.

Decision Variables - $y_i = \begin{cases} 1, & \text{if hospital } \mathrm{h}_i \text{ is selected} \\ 0, & \text{if hospital } \mathrm{h}_i \text{ is not selected} \end{cases}$

Objective Function - $\sum_{i=1}^{k} y_i$ where k is the number of hospitals present. We wish to minimize the objective function since we want the minimum number of hospitals to be active.

Constraints -

1. $\sum_{i|h_i \in H_j} y_i \geq 1 \; \forall j \in 1, 2, \ldots, m$ where h_i denotes the hospital i, $1 \leq i \leq m$ and H_j is the set of all hospitals that cater to the cell a_j. Note that in this case, $n = 10$.

2. $y_i \leq 1 \; \forall i \in \{1, 2, \ldots, k\}$

3. $y_i \geq 0 \; \forall i \in \{1, 2, \ldots, k\}$

Constraint 1 denotes that for every cell, $\sum y_i$ of all the hospitals covering that cell should be greater than or equal to 1. This in effect takes into account that each cell is being covered by at least 1 hospital. This formulation was coded and run in AMPL and the results obtained are discussed in section 5.

3.2 Modified Case

In this case, we further take into account that each hospital has a capacity and each cell has a population density associated with it. We tackle a preliminary case where the capacity of each hospital and the population density of each cell is constant. These new parameters of capacity and density are denoted by **c** and **d** respectively.

Decision Variables - $y_i = \begin{cases} 1, & \text{if hospital } \mathrm{h}_i \text{ is selected} \\ 0, & \text{if hospital } \mathrm{h}_i \text{ is not selected} \end{cases}$

Objective Function - $\sum_{i=1}^{k} y_i$ where k is the number of hospitals present. The decision variable and the objective function remain the same as in the previous case.

Constraints -

We introduce a new variable α_{i_j} which denotes the fraction of population of cell a_i that hospital h_j will be to be catering to. This consideration is important since because of a constraint on the capacity of hospitals, different portions of populations from a cell might be catered by different hospitals. (This is given that the cell is in the area of coverage of more than one hospitals).

1. $0 \leq \alpha_{i_j} \; \forall i \in \{1, 2, \ldots, n^2\}, j \in \{1, 2, \ldots, m\}$ where n^2 is the number of cells and m is the number of hospitals.

2. $0 \geq \alpha_{ij} \; \forall i \in \{1, 2, \ldots, n^2\}, j \in \{1, 2, \ldots, m\}$

3. $\sum_{i=1}^{n^2} \alpha_{i_j} \dot{d} \leq c \ \forall j \in \{1, 2, \ldots, m\}$

4. $\sum_{j|h_j \in H_i} \alpha_{i_j} \geq 1 \ \forall i \in \{1, 2, \ldots, n^2\}$ where h_j denotes the hospital j, $1 \leq j \leq m$ and H_i is the set of all hospitals which cater to cell a_i.

5. $\sum_{i=1}^{n^2} \alpha_{i_j} \leq 100 y_j \ \forall j \in \{1, 2, \ldots, m\}$

Constraints 1 and 2 imply that α_{i_j} should be between 0 and 1. Constraint 3 is the summation over all the cells for each hospital. The total population catered by each hospital should be less than the capacity of the hospital. Constraint 4 essentially takes into account that no portion of any cell is remaining uncovered. This constraint is similar to Constraint 1 in the Base Case of the problem. A hospital h_j will be inactive only if $\alpha_{i_j} = 0 \ \forall i \in \{1, 2, \ldots, n^2\}$ for that particular j. Hence to finally assign y_j as 0 or 1 we have Constraint 5. If $\sum_{i=1}^{n^2} \alpha i_j = 0$ then according to Constraint 5, y_j will be 0, and 1 otherwise. Note that constraints 4 and 5 essentially capture the equivalent nonlinear constraint : $\sum_{j=1}^{m} \alpha_{i_j} y_j \geq 1 \ \forall i \in i, 2, \ldots, n^2$

This formulation with an added capacity constraint was run in AMPL and the results are discussed in section 5. Further, different population densities of all cells were used for a constant capacity of hospitals and a comparative study was done.

4 Greedy Algorithm

Often, optimization problems with a large number of decision variables, parameters, and constraints end up requiring significantly large computational times. There is often not a hard requirement of needing the optimal solution and tolerances allow for slight deviations from optimum. In the context of our problem, if we are looking at expanding our model to very large grids, we would be faced with equivalently large computational times. Further, having a few more hospitals than optimum wouldn't put such a large strain on the resources. In this context, we want to check the performance of the greedy algorithm (a heuristic) and compare it to the results of the CPLEX solver.

For this, a code implementation was done using Python. The Greedy Algorithm is an iterative procedure which selects the best existing hospital one at a time, till all the cells have been exhausted. Initially, you start by assuming none of the cells are covered. You check which hospital has the most number of not covered cells in its neighborhood. This hospital is then selected and all the cells in its domain have their status changed to 'covered'. A flow chart for the algorithm is presented in Fig. 3. This algorithm was run for several cases, and the results are tabulated in section 5

5 Results

5.1 Base and Advanced Case Comparison

The random assignment of hospitals in each cell with a probability of 40% was done using MATLAB. Using the CPLEX solver in AMPL alone, we could tackle a basic and a complex version of the problem. In case of a capacity of 1000 for each hospital and a population density of 100 in each cell in the modified case, we compare the results with the base case for 10x10 grid in Fig. 2.

In general, we can anticipate that adding capacity constraints would that mean more hospitals would be required. The table shown in Fig 4. captures this trend which is observed in grids of sizes 10x10 to 15x15 having a distribution of hospitals with 40% probability in each cell. It was observed that the location of the active hospitals with and without capacity was significantly different. This will become important in section 6, when we look at the robustness of the solution to slight changes in the parameters of the problem.

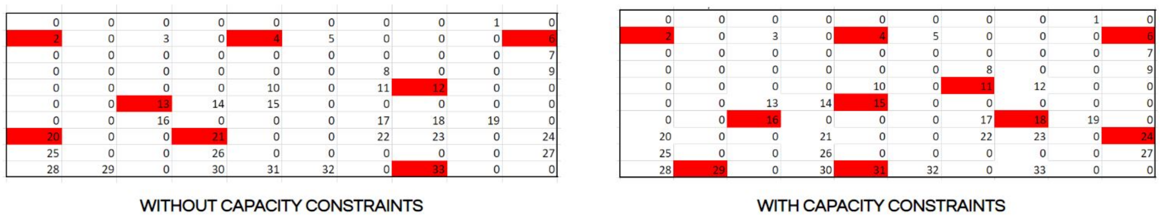

Figure 2: The red cells demarcate the active hospitals. Without capacity constraints, only 8 hospitals need to be active out of 33, whereas assigning a capacity constraint increases the number of hospitals to 10.

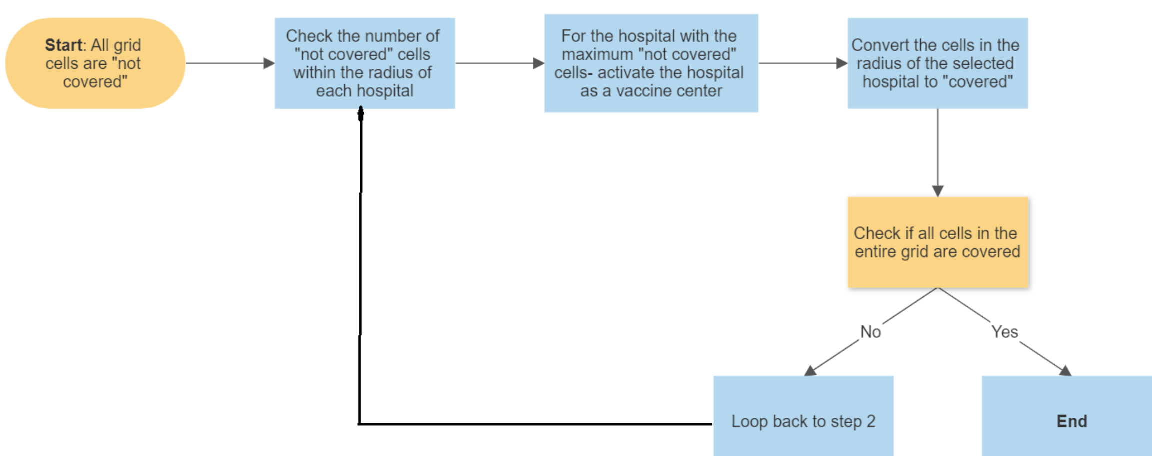

Figure 3: The Greedy Algorithm

Dimensions	10x10	11x11	12x12	13x13	14x14	15x15
Total cells	100	121	144	169	196	225
Total number of hospitals	33	39	49	58	70	85
Active hospitals with capacity	10	13	15	17	20	23
Active hospitals without capacity	8	9	9	10	11	14

Figure 4: In each case more hospitals are required with capacity as compared to without capacity.

5.2 Greedy Algorithm

Both the AMPL code and the Greedy Algorithm were run for 6 cases in total: starting from a 10x10 grid and ending at a 15x15 grid. Not just the number of hospitals were compared, but also which hospitals were selected. Not only were we getting a difference in the total number of selected hospitals, but also in the location of these selected hospitals. Further, it was observed that cplex performs significantly better than the greedy algorithm. The difference between these two is likely to blow up as we increase the parameters of the problem. For low number of parameters, the results are comparable. Fig. 5 displays the tabulated results.

What we conclude from this study is the fact that we should use greedy only when there is a requirement of very low computational time and no requirement of optimal solution. For CPLEX, the time blows up with large number of parameters and constraints, while for greedy, the errors blow up. We must make a trade-off based on our requirement, and prioritise either the error or the time.

6 Sensitivity Analysis

Optimal solutions have a tendency to be highly sensitive to small changes in the working parameters and noise. In our problem we have assumed population density to be constant (= 100) in each cell. In reality, population density is generally measured based on the Census data, which itself has probabilities of errors. This is because the method of conducting the census involves sampling of the population in regions and using statistics to get the best estimate of the population. Hence in reality, we could expect to have an actual population density of say, e.g. 150 instead of 100. Thus, considering the variation in the population density of cells we find the optimal locations of vaccination centres. As the optimal location configuration of vaccination centres changes, setting up of new centres and shutting down the old ones involves a cost. Hence we would prefer to have minimal changes in the optimal configuration with the change in population density.
Let cost of setting up a new vaccination centre be C_1 and the cost of shutting down an existing center be C_2.

For a population density of P_1 the optimal configuration has n_1 active hospitals. As P_1 changes to P_2 the new optimal configuration has say n_2 active hospitals. Let there be N common hospitals in these two optimal configurations. We define Number of Replacements (ρ) as $n_1 - N$ since each hospital of these $n_1 - N$ hospitals has to be shut down and a new one comes up for it. Higher ρ implies a larger change in the configuration of the optimal solution. We increase the population from 100 to 200 and observe the variation in ρ with respect to the previous population density. **Note -**

Case	Total	Greedy	Cplex
10	33	8	8
11	39	10	9
12	49	10	9
13	58	13	10
14	70	14	11
15	85	16	14

Figure 5: Table of results that shows the total number of hospitals, and the selected hospitals by cplex and greedy, for cases 10x10 through 15x15

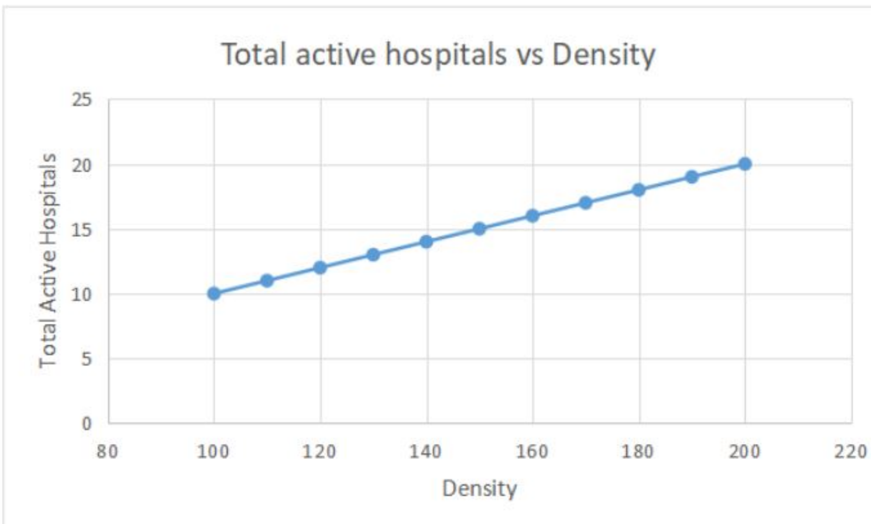

Figure 6: As population density increases by 10, the number of active hospitals increases by 1 at each step.

1. ρ is always a measure with respect to 2 population densities, the new one and the old one.
2. C_1 and C_2 are constant costs for the purpose of this project, while the capacity of each hospital is considered to stay constant at 1000 while the population density varies.
3. Total Cost involved due to change in the population density $= C_1(n_1 - N) + C_2(n_2 - N) = C_1(n_1 - N) + C_2(n_2 - n_1 + n_1 - N) = \rho(C_1 + C_2) + C_2(n_2 - n_1)$. Hence the total cost is directly proportional to ρ. Hence, we observe the variation of ρ as P_1 and P_2 change.

Figure 6 shows how total number of active hospitals change from 100 to 200, which as we expect increase as the population density increases. Figure 7 captures the trend in ρ with the increase in population density.

Conclusions : We thus observe that with the increase in population density in each cell, the number of active hospitals increases as expected. Note that these trends were observed in case of a 10x10 grid and we expect to observe similar trends in case of other dimensions of the grid.

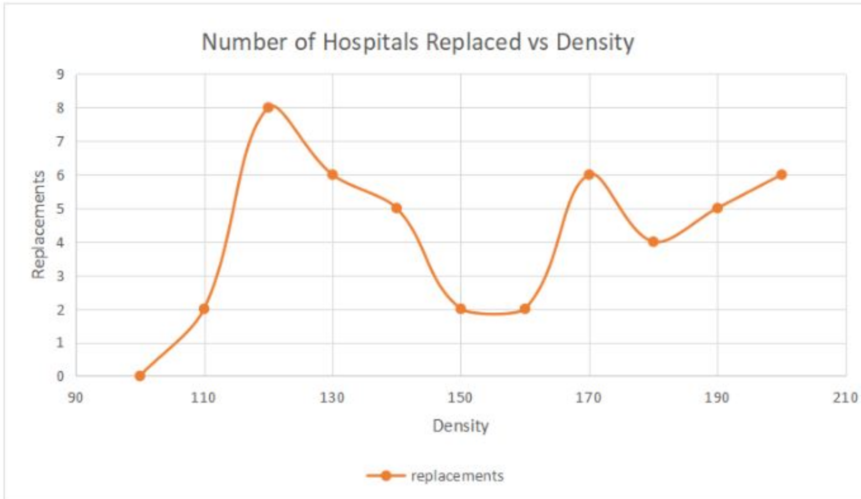

Figure 7: At some population densities, there is a sharp change in ρ whereas at some densities the graph is flatter

We observe that for a density of 120, there is a large change in rho if we move to either side of 120, which means that the optimal configuration at 120 is highly sensitive to change in density. On the other hand, a relatively flatter curve is observed near a density of 150, which indicates that not much change in the location of hospitals is needed for small changes in the population density.
Note that we cannot exceed a population density of 330 in each grid since the total population then would exceed the total capacity of all hospitals. However, with capacity constraints, we observe that for a population density in the range of 321 to 330, we need all hospitals to be active. Thus in case we have all hospitals active, we can handle a maximum of +3% increase in the population density.

7 Future Work and Improvements

With this work, we could explore the ways to handle a basic location optimization problem in an ideal model of a city. With further considerations of changes in capacity, changes in demand for vaccination and different population density in different locations, this work has a further scope of handling more realistic problems.

References

https://math.mit.edu/ goemans/18434S06/setcover-tamara.pdf