Social Distancing

Group-17

Introduction

In this project, we have studied the gains of social distancing in a city.

- Aim is to lower the contacts so that the number of vulnerable persons is small.
- Studying the effect of limiting the service providers' interactions in several ways.
- applying Graph algorithms in the real world scenario...

Assumptions

- In a Town, three different population levels are established for this project - 2000, 5000, 10000.
- Two separate levels of service providers are used - The first level assumes that the town has very few providers – 3% of the town population and Second level assumes the number of service providers to be 8% of the population.
- There will be some initially set infected people in the town and randomly set interactions among people of town.

Various Social Distancing Scenarios - Limiting the Interactions

- Without any social distancing
- With social distancing reducing contacts to 50%
- With social distancing reducing contacts to 33%
- With social distancing reducing contacts to 50% with assigned service provider - the odd numbered service provider will only interact with the persons with odd indices. Similarly, an even indexed person only gets service from a service provider with even index.

Creating the town and interactions among the people

- First, we create an adjacency matrix (Town matrix) for the town population
- randomly mark the required number of nodes in the matrix as service providers
- set randomly the initially infected persons - `randomYes()`
- Service providers are also included in the adjacency matrix and their interaction is also set in the matrix.

RandomYes() and BiasedYes()

- *randomYes()* - returns 1 only 0.1% of the time and returns 0, for 99.9% of the time to randomly infect some persons and to select service providers.
- Interactions among the populations were not random.
- *BaisedYes()* - to introduce some bias.
- Count the prime factors common in two indices of Adjacency matrix.
- More prime factor they have in common, more often we call *RandomYes()*

Modified Floyd Warshall - Transitive closure

- for letting the population interact i.e. we have found the transitive closure of that graph of population.

```
void transClosure(int population)
{
    for (int k = 0; k < population; k++)
        for (int i = 0; i < population; i++)
            for (int j = 0; j < population; j++)
                town[i][j] = town[i][j] || (town[i][k] && town[k][j]);
}
```

Conclusion

- Social distancing does have an impact on controlling cases.
- Number of Infected persons decreases with limiting interactions.
- Restricting service provider doesn’t mean much until we restrict normal residents to an extreme extent
- Faced challenges while deciding the various setting of the town, normal residents’ interactions, service providers’ interactions in accordance with the total population.
- Solved it by sampling different inputs and testing from multiple cases and got the most suitable settings

Thanks!

Abhishek Kumar - 19UCS241
Jaskaran Singh - 19UCS262
Raja Sethi - 19UCS239
Saurabh Shukla - 19UCS260
Sidharth Akar - 19UCS215