Vrishank Jannu

vgj95

COE 322 FINAL PROJECT

INFECTIOUS DISEASE SIMULATION

12/7/21

Introduction

With the recent COVID-19 pandemic, the study of epidemiology and the modeling of infectious diseases has soared in over the past several months. Public health officials and medical practitioners and leaders are all looking for innovative ways to be one step ahead in the fight to end diseases. This project simulates an infectious disease and displays how a disease spreads and gets eradicated in a population. It uses the size of a population, the probabilities of individuals getting vaccinated and infected, as well as the number of days any individual is infected (severity of the disease) to model the spread and eventual eradication of a disease. The goal of this project is to ultimately show how taking certain steps like getting vaccinated can reduce the overall transmission of disease in a population and speed the process of ending the disease.

Coding Process & Explanation

Below is my entire Infectious Disease Simulation file and its code:

```
// Vrishank Jannu || vgj95 || COE 322 FINAL PROJECT
#include <iostream>
#include <vector>
#include <memory>
#include <iomanip>
#include <string>
#include <random>
using namespace std;

class Person {
private:
    int num_days;
    int state = 0;
    string status;
public:
    Person() {};
    int update() {
        if (state > 1) {
            state = state - 1;
            return state;
        }
        if (state == 1) {
            state = state - 2;
            return state;
        }
    }
    void infect(int num_days) {
        if (state == 0) {
            state = num_days;
```

```
        }
    }
    string status_string() {
        if (state == 0) {
            status = "susceptible";
            return status;
        }
        else if (state == -1) {
            status = "recovered";
            return status;
        }
        else if (state == -2) {
            status = "vaccinated";
            return status;
        }
        else if (state > 0) {
            status = "sick (" + std::to_string(state) + " days to go)";
            return status;
        }
    };
    bool is_stable() {
        if (state == -1) {
            return true;
        }
        else {
            return false;
        }
    }
    int get_state() {
        return state;
    }
};

class population : Person {
private:
    int counter;
    int randnum;
    float infect_prob;
    int vaccinated;
    string popn;
public:
    vector<Person> people;

    int days;
    int npeople;
    population() {};
    population(int npeople) : npeople(npeople)
    {
        people = vector<Person>(npeople);
    }
    int num_days(int num) {
        days = num;
        return days;
    }
    void random_infection() {
        randnum = rand() % people.size();
        return people[randnum].infect(days);
    };
    void random_vaccination() {
        //randnum = rand() % people.size();
        // return people[randnum].infect(-2);
```

```
        for (int i = 0; i < people.size(); i++) {
            float bad_luck = ((float)rand() / (float)RAND_MAX);
            if (bad_luck > .95) {
                people[i].infect(-2);
            }
        }
    }
    int count_infected() {
        counter = 0;
        for (int i = 0; i < people.size(); i++) {
            if (people[i].get_state() > 0) {
                counter++;
            }
        }
        return counter;
    }
    void update2() {
        for (int i = 0; i < people.size(); i++) {
            people[i].update();
        }
    }


    void popn_status() {
        for (int i = 0; i < people.size(); i++) {
            if (people[i].get_state() > 0) {
                cout << "+";
            }
            else if (people[i].get_state() == 0) {
                cout << "?";
            }
            else if (people[i].get_state() == -1) {
                cout << "-";
            }
            else if (people[i].get_state() == -2) {
                cout << "V";
            }
        }
    };
    void probability_of_transfer() {
        for (int i = 0; i < people.size(); i++) {
            float bad_luck = ((float)rand() / (float)RAND_MAX);
            if (bad_luck > .95 && (i - 1) > 0 && (i + 1) < people.size()) {
                people[i - 1].infect(days);
                people[i + 1].infect(days);
            }
        }
    }
};

    int main() {
        srand(static_cast <unsigned> (time(0)));
        int step = 1;
        int days_total, citypop;
        cout << "Please enter population of the city: " << endl;
        cin >> citypop;
        population city(citypop);
        cout << "Please enter the number of days: ";
        cin >> days_total;
        city.num_days(days_total);
        city.random_infection();
```

```
        while (city.count_infected() > 0) {
            city.random_vaccination();
            city.update2();
            int count = city.count_infected();
            cout << "There are " << count << " people infected on Day " << step << ".
Simulation:";
            city.popn_status();
            city.probability_of_transfer();
            step++;
            cout << endl;
        }
        return 0;
    }
```

First, I did the includes. Next, I defined a "Person" class which essentially includes all of the basic information for each individual in the population. In the Person class, the infect function displays whether or not a person is infected. The update function updates the day each time. The status_string function shows the status of the individual whether he is recovered, vaccinated, or susceptible. The get_state function simply gets the state of the individual and the is_stable boolean function indicates whether the individual is sick or not. Next, I proceeded onto the population class. In this class, the random_infection method randomly chooses the first person to be infected whereas the probability_of_transfer function which is developed later allows for individuals to randomly infect others. The popn_status indicates the status of everyone in the population. The update_2 function operates the exact same way as the update() function in the Person class. Lastly, I modified the main function to incorporate my methods and functions and properly display the outputs.

```
Please enter population of the city:
50
Please enter the number of days: 5
There are 1 people infected on Day 1. Simulation:????????????+?????????????????V?????????????????V?
There are 6 people infected on Day 2. Simulation:??????V?????+??+?+?????+?+??+?V?????????????????V?
There are 6 people infected on Day 3. Simulation:??V???V?????+??+?+?????+?+??+?V???V?????????????V?
There are 10 people infected on Day 4. Simulation:??V?+?V???+?+??+?+V????+?+??+?V???V???????????V+V+
There are 10 people infected on Day 5. Simulation:??V?+?V???+V-??+?+V????+?+??+?V?+?V???????????V+V+
There are 8 people infected on Day 6. Simulation:??VV+?V???+V-??-?-V?++?-?-??-?V?+?V???V??????+V+V+
There are 12 people infected on Day 7. Simulation:??VV+VV???+V-??-?-V?+++-?-??-?V?+?V?+?V??+?+V+V+V+
There are 10 people infected on Day 8. Simulation:??VV-VV??V-V-??-?-V?+++-?-??-?V?+?V?++V+?+?+V+V-V-
There are 12 people infected on Day 9. Simulation:V?VV-VV??V-V-?V-?-V?+++-+-+?-VV?-?V?++V+?+++V+V-V-
There are 11 people infected on Day 10. Simulation:V?VV-VV+?V-V-?V-?-V+--+-+-+?-VV?-?V?++V+V+++V-V-V-
There are 8 people infected on Day 11. Simulation:V?VV-VV+?V-V-?V-?-V+----+-++-VV?-?V?-+V+V-+-V-V-V-
There are 6 people infected on Day 12. Simulation:V?VV-VV+?V-V-?V-?-V+----+-++-VV?-?V?--V-V-+-V-V-V-
There are 3 people infected on Day 13. Simulation:V?VV-VV+VV-V-?V-?-V+-------+-VV?-?V?--V-V---V-V-V-
There are 1 people infected on Day 14. Simulation:V?VV-VV-VV-V-?V-?-V--------+-VV?-?V?--V-V---V-V-V-
There are 0 people infected on Day 15. Simulation:V?VV-VV-VV-V-?V-?-V----------VV?-?V?--V-V---V-V-V-

C:\Users\15128\source\repos\COE322Final\Debug\COE322Final.exe (process 16148) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close
le when debugging stops.
Press any key to close this window . . .
```

This figure above represents an example of the output. The population size is 50 individuals and the number of days is 5. The probabilities of both transmission and vaccination were both 5%. It displays the number of individuals infected each day and displays the correct symbol. The question mark indicates that an individual is susceptible. We can see how a handful of individuals were susceptible throughout the entire simulation which displays the concept of "Herd Immunity".

Data Gathering Process

I ran a total of 8 studies, each consisting of 10 trials. In each trial, the same numbers were inputted into the program 3 total times to ensure better accuracy in the data and reduce the randomness effect. In other words, the total number of days for Disease to End (which is the output) was taken 3 times and the average of the three values is what is displayed in the last column of each table.

Each of these studies contains 4 input variables:

1) Number of People
2) Number of Days that someone is infected
3) Probability that a Random Individual in the population is vaccinated.
4) Probability that a Random Individual in the population transfers the disease.

To examine the relationships and behaviors in the simulation, we will do two things:

1) Modify at least one input to reflect an increasing or decreasing value as the # of trials increases.
2) Identify and analyze the total number of days that the disease lasted (the output) after each trial.

I would also like to elaborate on the modifications I made in the program during each trial to get the outputs. In Line 93 of the "random_vaccination" void function, I modified the threshold decimal in the if statement. For example, if we want the probability of getting vaccinated to be 0.85 in the trial, the threshold and value in the if statement will be modified to 0.15. In other words, the equation for finding the value to enter in the program for the simulation is shown below:

$$Desired\ Probability = 1 - (Value\ entered\ in\ the\ program)$$

This same equation to determine what to enter in the program for the probability of randomly infecting an individual in the population applies in Line 133 of the "probability_of_transfer" void function.

Data Results and Analysis

Study #1: Increasing Recovery Times

Trial	Population	# Days for Infection	Probability of Getting Vaccinated	Probability of Infecting	# Days for Disease to End
1	50	2	0.15	0.15	8
2	50	4	0.15	0.15	14
3	50	6	0.15	0.15	15
4	50	8	0.15	0.15	14
5	50	10	0.15	0.15	22
6	50	12	0.15	0.15	20
7	50	14	0.15	0.15	27
8	50	16	0.15	0.15	23
9	50	18	0.15	0.15	26
10	50	20	0.15	0.15	33

In this study, the variable that is being changed is the number of days for a person to stay infected. We kept the probability of getting infected and vaccinated low (at 15%). We also kept the population of the city the same at 50 people. As we can see in the last column, the total number of days increases over time as the number of days that an individual is infected increases which resembles a positively correlated relationship. Increasing recovery times results in increasing disease eradication times.

Study #2: City Expansion

Trial	Population	# Days for Infection	Probability of Getting Vaccinated	Probability of Infecting	# Days for Disease to End
1	20	10	0.15	0.15	19
2	40	10	0.15	0.15	19
3	60	10	0.15	0.15	17
4	80	10	0.15	0.15	25
5	100	10	0.15	0.15	17
6	120	10	0.15	0.15	19
7	140	10	0.15	0.15	20
8	160	10	0.15	0.15	30
9	180	10	0.15	0.15	23
10	200	10	0.15	0.15	22

In this study, the input variable that is being changed is the population. We increase it linearly as we proceed through the trials. We can see that the population size does not have a significant effect on the number of days it takes for the disease to end especially if the probability of getting vaccinated and infected is significantly low.

Study #4: Inconclusive on Increasing Infection Chances

Trial	Population	# Days for Infection	Probability of Getting Vaccinated	Probability of Infecting	# Days for Disease to End
1	50	10	0.15	0.05	21
2	50	10	0.15	0.15	17
3	50	10	0.15	0.25	17
4	50	10	0.15	0.35	15
5	50	10	0.15	0.45	13
6	50	10	0.15	0.55	14
7	50	10	0.15	0.65	12
8	50	10	0.15	0.75	12
9	50	10	0.15	0.85	11
10	50	10	0.15	0.95	10

In this study, the variable that is being modified is the probability of infecting. As the transmission probability goes up, the number of days for the disease to end appears to decrease slowly over time if the vaccination probability remains the same.

Studies #3, & #5-8: Importance of Vaccinations

Trial	Population	# Days for Infection	Probability of Getting Vaccinated	Probability of Infecting	# Days for Disease to End
1	50	10	0.05	0.15	22
2	50	10	0.15	0.15	16
3	50	10	0.25	0.15	17
4	50	10	0.35	0.15	15
5	50	10	0.45	0.15	16
6	50	10	0.55	0.15	10
7	50	10	0.65	0.15	12
8	50	10	0.75	0.15	11
9	50	10	0.85	0.15	11
10	50	10	0.95	0.15	10

In this study, the variable that is being changed is the probability of getting vaccinated. We kept the probability of infecting the same (at 15%). We can see that the total number of the days for the disease to be eradicated decreases over time as the probability of getting vaccinated increases. This study displays the importance of vaccinations and why getting a vaccine saves lives and reduces disease transmission.

Trial	Population	# Days for Infection	Probability of Getting Vaccinated	Probability of Infecting	# Days for Disease to End
1	50	5	0.05	0.05	30
2	50	5	0.15	0.15	17
3	50	5	0.25	0.25	18
4	50	5	0.35	0.35	14
5	50	5	0.45	0.45	12
6	50	5	0.55	0.55	13
7	50	5	0.65	0.65	13
8	50	5	0.75	0.75	11
9	50	5	0.85	0.85	11
10	50	5	0.95	0.95	8

In this series of trials, we can also see that the number of days for the disease to be fully eradicated in the population decreases as the probability of getting vaccinated increases. Despite the probability of disease transmission also increasing over time, the probability of vaccinations ultimately holds higher importance in eradicating the disease.

Trial	Population	# Days for Infection	Probability of Getting Vaccinated	Probability of Infecting	# Days for Disease to End
1	50	5	0.95	0.95	7
2	50	5	0.85	0.85	6
3	50	5	0.75	0.75	9
4	50	5	0.65	0.65	11
5	50	5	0.55	0.55	10
6	50	5	0.45	0.45	14
7	50	5	0.35	0.35	17
8	50	5	0.25	0.25	16
9	50	5	0.15	0.15	18
10	50	5	0.05	0.05	20

Alternatively, we can see how both probabilities decreasing still displays the importance of getting vaccinated. Even though the disease transmission probability decreases over time, the number of days for the disease to be fully wiped out still rises over time because the number of individuals in the population getting vaccinated continues to get lower.

Trial	Population	# Days for Infection	Probability of Getting Vaccinated	Probability of Infecting	# Days for Disease to End
1	50	5	0.95	0.05	10
2	50	5	0.85	0.15	11
3	50	5	0.75	0.25	12
4	50	5	0.65	0.35	14
5	50	5	0.55	0.45	14
6	50	5	0.45	0.55	17
7	50	5	0.35	0.65	16
8	50	5	0.25	0.75	19
9	50	5	0.15	0.85	22
10	50	5	0.05	0.95	21

In the next two studies, I increased one of the probabilities while decreasing the other. In this study above, we can see that the disease eradication time significantly increases over time as the probability of getting infected increases and the probability of getting vaccinated decreases over time. Green represents an increasing probability over time. Red represents a decreasing probability over time. In the study below, we can see that once again, the number of people receiving vaccinations in the population increasing results in decreasing disease eradication times. Additionally, in the study below, the number of “?”’s outputted was much higher over time as the probability of getting vaccinated increases. From this, we can infer that the higher the vaccination probability, the faster a population reaches herd immunity.

Trial	Population	# Days for Infection	Probability of Getting Vaccinated	Probability of Infecting	# Days for Disease to End
1	50	5	0.05	0.95	15
2	50	5	0.15	0.85	12
3	50	5	0.25	0.75	12
4	50	5	0.35	0.65	10
5	50	5	0.45	0.55	11
6	50	5	0.55	0.45	10
7	50	5	0.65	0.35	8
8	50	5	0.75	0.25	7
9	50	5	0.85	0.15	6
10	50	5	0.95	0.05	6

Conclusion

Based on the results and analysis, we can conclude that the size of the population and the disease transmission probability (probability of infecting) hold a much lower weight in affecting the time for a disease to fully be eradicated. However, we can see that no matter how large or small the population may be or how severe the disease may be (# of days that an individual is infected), the probability of getting vaccinated holds a significantly heavy weight in the time it takes for the disease to be wiped out. We can conclude that a higher vaccination rate in a population will result in a reduction in disease transmission and a faster disease eradication time.