Introduction

We designed the virus transmission simulation based on the SEIR (Susceptible-Exposed-Infective-Recovered). The model depends on various parameters that characterize the virus, population, and residence. As these parameters are expected to vary significantly, our project can be adjusted with the dynamic population, virus, and residence-dependent parameters.

Project Goal

We are deterministic to find the persistence of the pandemic considering control measures. This study of incorporating the SEIR model to covid-19 dynamics with the pathogen and resident parameters helps to get the reproduction rate (R-factor) and Dispersion rate (K – factor). We follow the stochastic approach generating random data to find these factors.

Modeling Details

We model the population interacting into four categories as stated below:

Susceptible (S): Susceptible individuals are those who were never been infected with the virus and have no immunity against virus. Susceptible individuals become exposed once they are infected with the disease.

Exposed (E): Those who have been infected but are not yet infectious themselves.

Infectious (I): Individuals present clinical symptoms and are infectious and can spread the virus to those who come into close contact. Usually, they are reported as confirmed cases.

Removed (R): Removed individuals include deceased patients and recovered ones who are immune to future infection from COVID-19.

$$\textit{Population } P = S + E + I + R$$

We considered the below properties for a person which are dynamic and dependent on the set of population.

1. Activity Rate: The rate of people goes out every day.

2. Activity Radius: This is the radius of the resident’s activities.

The properties of the pathogen can be dynamic which includes in the Virus class:

1. Infectious Radius: The radius within which the person can get infected.
2. Infectious Rate: The probability at which the disease can be transmitted between susceptible and infectious people.
3. Removal Rate: This is the mortality rate for an infected person.
4. Recovery Rate: Recovery rate is the rate at which the infected person can recover.
5. Incidence Rate: This is the rate at which the exposed person becomes infective.
6. Infectious Period: The average number of days a person is infectious.

We considered the effectiveness of measures in Measure class as a significant property which vary and determine the persistence and effect of the epidemic.

1. Virus Testing: The number of tests performed per day.
2. Vaccine: Usage of number of vaccines per day.
3. Mask use rate and Mask effectiveness: These parameters define the usage rate and its effectiveness of the mask.
4. Quarantine rate: The probability at which each infected person obeys the quarantine rule.

We defined the reproduction rate R factor, as the ratio of the exposed residents to the infected residents

$$R = E/I$$

We calculate the K factor by summing up the number of exposed and infected residents and check with the infection rate and take the ratio of the total residents who spread with the sum.

Implementation and Design

The project has multiple functions defined to calculate infected people, spread of virus, measure of vaccine and testing and calculate the R and K factor.

Getting the parameters of Virus, Measure and Residence from the properties file, we then generate the residents and the initially infected people in the constructor of residence. Location and initial status of a person are defined.

1. Spread of Virus

   Within the scope of all the residents, we take two people and determine if either of them is contagious. Then we determine the distance between two people. If the distance is less than the infectious radius then we'll have the other person infected.

```
If (p1 is contagious || p2 is contagious)
If (distance < infectious radius)
        Infect (p1 or p2)
```

2. Infection

   Considering the stochastic random generation process here, we take the value and compare with the mask usage rate and mask effectiveness rate. If both of the randomly generated values are less than the mask measure parameters and infectious rate, we consider the person infected.

```
If (random < mask usage || random < mask effective) return
If (random < infectious rate) set person as infected.
```

3. Update resident status

   We update the status of the residents by considering the person status. We categorize the count of people and update the susceptible, exposed, infected, and removed if the person is recovered. If the randomly generated value is greater than the recovery rate, then we update the count of the dead residents.

4. Calculate RK

   We sum the exposed and infected count for each day. The reproduction rate, R – factor is calculated as the ratio of the exposed to the infected residents. We calculate the dispersion rate, K – factor as the ratio of the top spreaders (count of people infected by) to the sum of exposed and infected.

$$R = dE/dI$$
$$K = \text{top spreaders}/E+I$$

5. Apply Measure:

   We then apply measures like vaccine and virus testing by keeping track of the number of vaccines produced per day and the vaccines given. If a person is susceptible, we give the vaccine and set the status of the person as removed while decrementing the vaccine count.

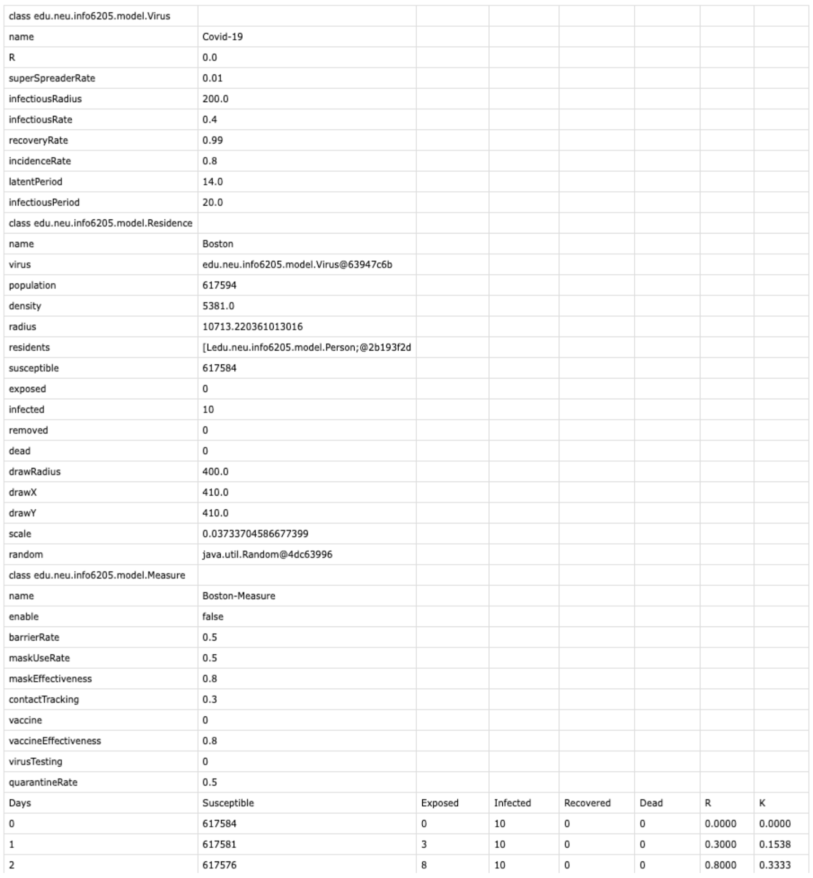
   If (person susceptible) set person status removed

We have the number of testing kits and when we test the person, we consider the person who is exposed and infected. If the randomly generated value is less than the quarantine rate from the measure parameter, we set the status of the person as removed.

```
If ((person is exposed or infected) and (random < quarantine rate))
        Set person status removed.
```

6. Export CSV

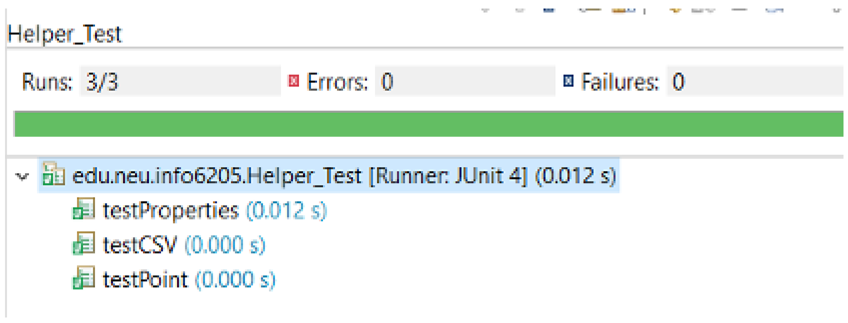
   Finally, if the count of exposed and infected is 0, we pump the results and logs to the csv file. We publish the number of days it takes for humans to fight over the persistence of the covid – 19 virus transmission. (CSV reports with different parameters were uploaded in the project)



class edu.neu.info6205.model.Virus							
name	Covid-19						
R	0.0						
superSpreaderRate	0.01						
infectiousRadius	200.0						
infectiousRate	0.4						
recoveryRate	0.99						
incidenceRate	0.8						
latentPeriod	14.0						
infectiousPeriod	20.0						
class edu.neu.info6205.model.Residence							
name	Boston						
virus	edu.neu.info6205.model.Virus@63947c6b						
population	617594						
density	5381.0						
radius	10713.220361013016						
residents	[Ledu.neu.info6205.model.Person;@2b193f2d						
susceptible	617584						
exposed	0						
infected	10						
removed	0						
dead	0						
drawRadius	400.0						
drawX	410.0						
drawY	410.0						
scale	0.03733704586677399						
random	java.util.Random@4dc63996						
class edu.neu.info6205.model.Measure							
name	Boston-Measure						
enable	false						
barrierRate	0.5						
maskUseRate	0.5						
maskEffectiveness	0.8						
contactTracking	0.3						
vaccine	0						
vaccineEffectiveness	0.8						
virusTesting	0						
quarantineRate	0.5						
Days	Susceptible	Exposed	Infected	Recovered	Dead	R	K
0	617584	0	10	0	0	0.0000	0.0000
1	617581	3	10	0	0	0.3000	0.1538
2	617576	8	10	0	0	0.8000	0.3333

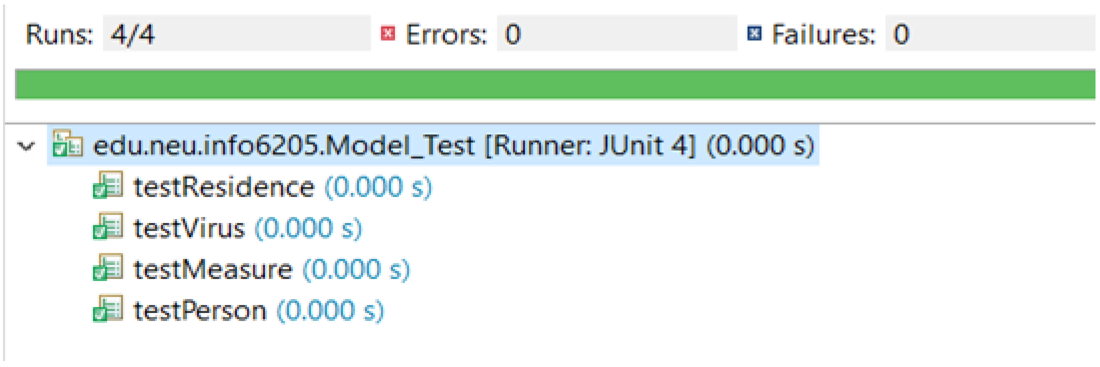
Test Cases

We initialized test cases to make sure that our simulation runs as expected. The run for these test cases is given below.



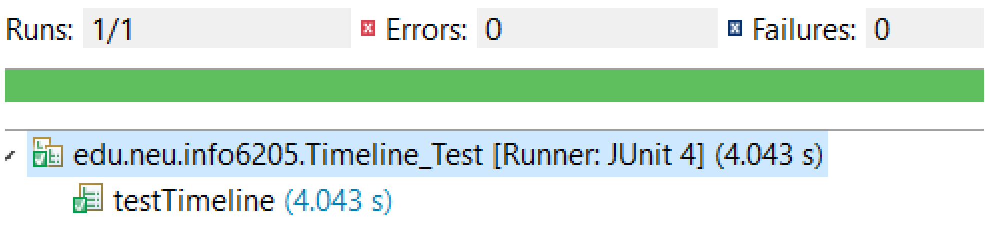
```
Helper_Test
Runs: 3/3        Errors: 0        Failures: 0

edu.neu.info6205.Helper_Test [Runner: JUnit 4] (0.012 s)
    testProperties (0.012 s)
    testCSV (0.000 s)
    testPoint (0.000 s)
```



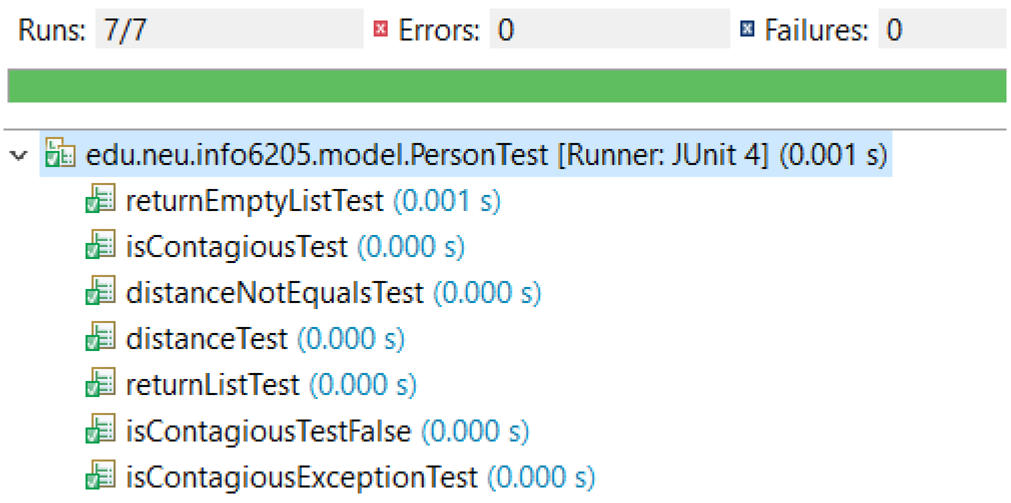
```
Runs: 4/4        Errors: 0        Failures: 0

edu.neu.info6205.Model_Test [Runner: JUnit 4] (0.000 s)
    testResidence (0.000 s)
    testVirus (0.000 s)
    testMeasure (0.000 s)
    testPerson (0.000 s)
```



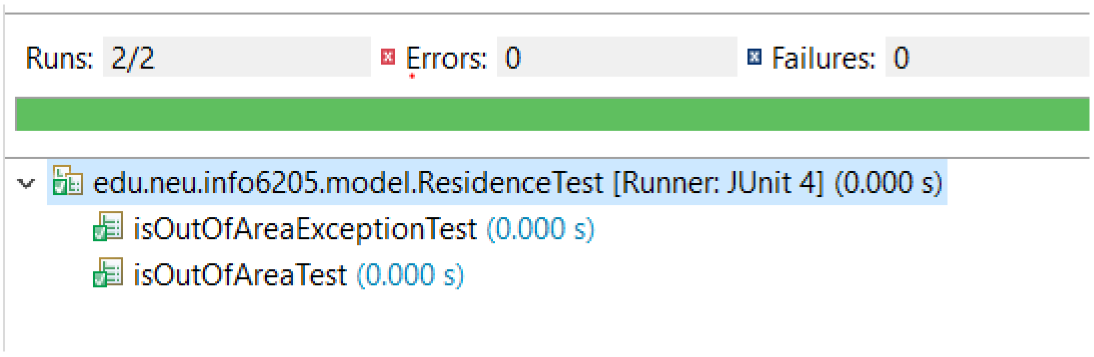
```
Runs: 1/1        Errors: 0        Failures: 0

edu.neu.info6205.Timeline_Test [Runner: JUnit 4] (4.043 s)
    testTimeline (4.043 s)
```



```
Runs: 7/7        Errors: 0        Failures: 0

edu.neu.info6205.model.PersonTest [Runner: JUnit 4] (0.001 s)
    returnEmptyListTest (0.001 s)
    isContagiousTest (0.000 s)
    distanceNotEqualsTest (0.000 s)
    distanceTest (0.000 s)
    returnListTest (0.000 s)
    isContagiousTestFalse (0.000 s)
    isContagiousExceptionTest (0.000 s)
```


```
Runs: 2/2    Errors: 0    Failures: 0

edu.neu.info6205.model.ResidenceTest [Runner: JUnit 4] (0.000 s)
    isOutOfAreaExceptionTest (0.000 s)
    isOutOfAreaTest (0.000 s)
```

Simulation Visualization

Introduction

The simulation of the Covid virus transmission is popped with an interesting GUI which updates stats of the simulation for each day. We use swing to visualize the whole process. Each dot represents one person, and the rectangle area represents the city area.

Design Details

1. Randomness

   In order to simulate people's random behaviors, we use Gaussian Function to generate people's moving path, and process other parameters to add randomness. For example:

   ```
   double dx = Person.activityRadius / 2.0 * random.nextGaussian();
   double dy = Person.activityRadius / 2.0 * random.nextGaussian();
   ```

   Using Person's activity radius multiple random.nextGaussian() gets person's random moving path.

2. Consistency

   Since we need to compare the results between different parameters. We need to control environmental parameters in different simulations. For our program, the most important environmental parameter is Random Function. In order to Improve consistency, we use a singleton pattern to create a static random instance with a constant seed.

   ```
   private static RandomUtil randomUtil = new RandomUtil();
   ```

```
private final Random random;
private RandomUtil(){
  random = new Random(31);
}

public static Random random(){
  return randomUtil.random;
}
```

3. Multi-threaded

   Running both the simulation process and drawing process need a pretty long running time. To increase the speed of the whole program, we use two threads to run these two parts of modules.

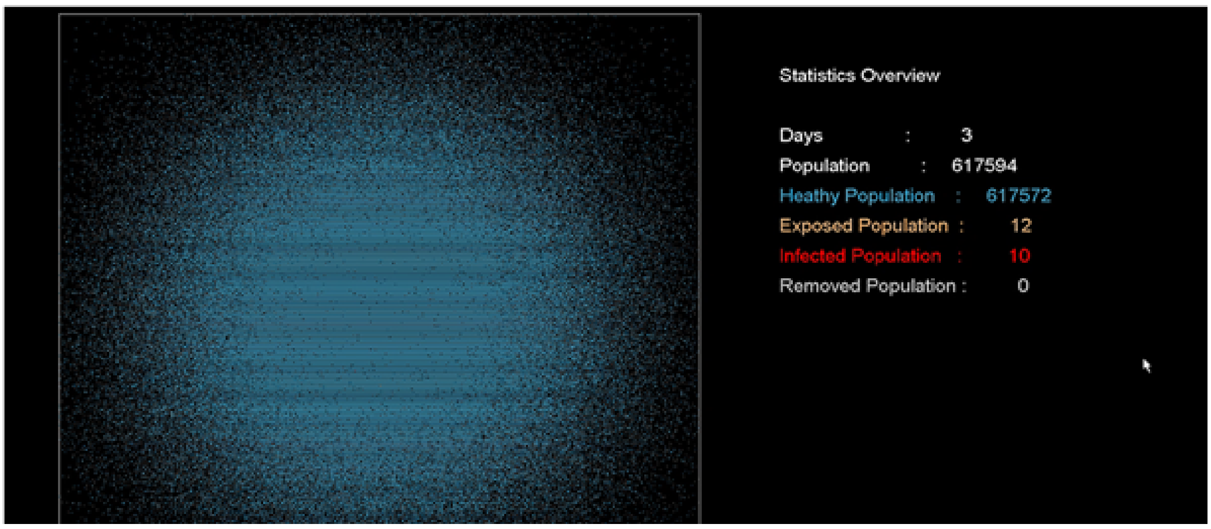
4. Mapping

   The size of JFrame is fixed, but we need to simulate different areas, from big cities to small towns. So we use multiple mapping functions to map the true distance and areas (mile or m) to the coordinate in JFrame.

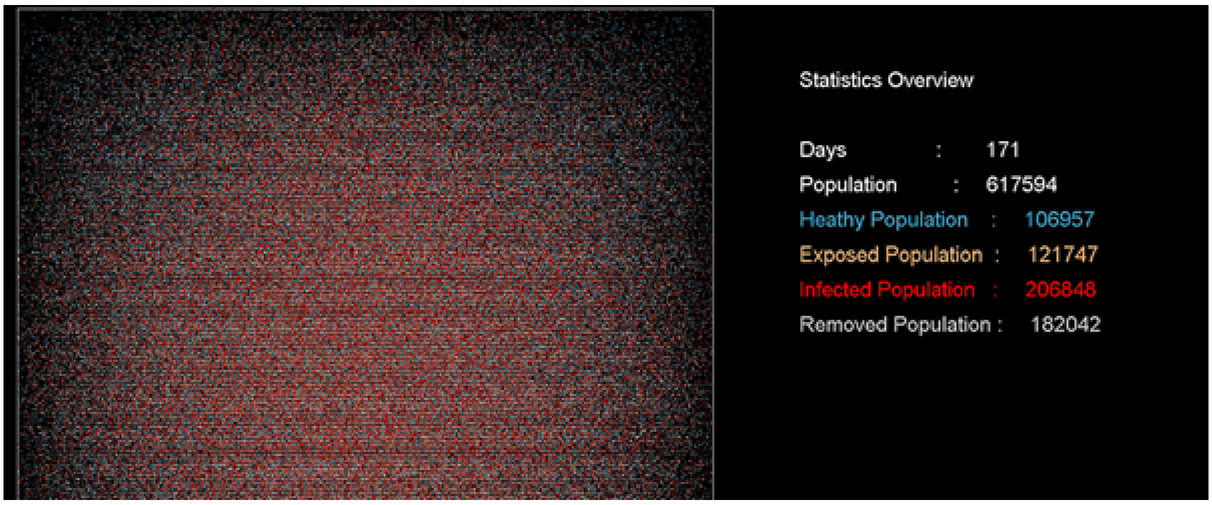
Simulation Video

click link to watch the simulation video:

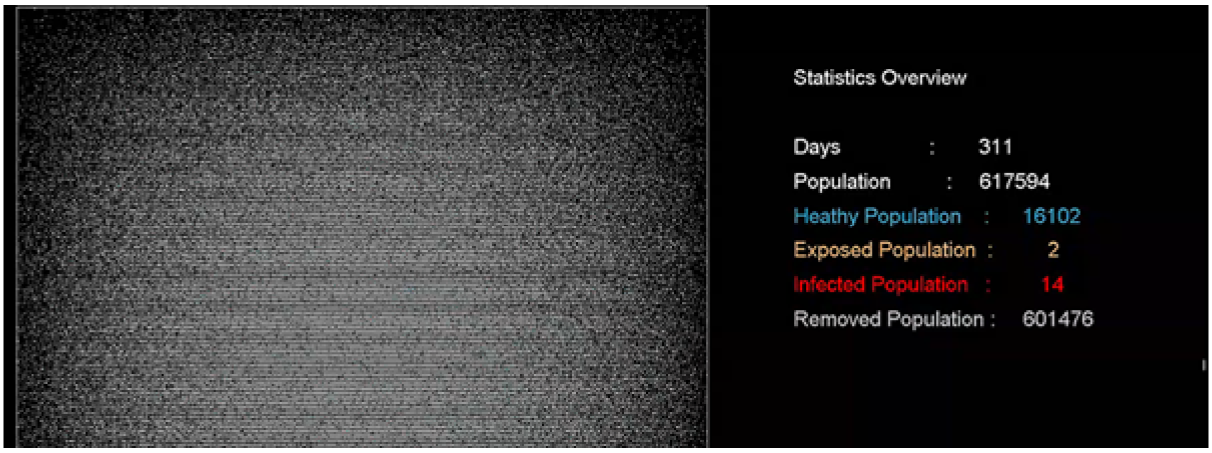
https://www.youtube.com/watch?v=8GAmKTwkzUw


```
Statistics Overview

Days                :      3
Population          :   617594
Heathy Population   :   617572
Exposed Population  :      12
Infected Population :      10
Removed Population  :       0
```


```
Statistics Overview

Days                :    171
Population          :   617594
Heathy Population   :   106957
Exposed Population  :   121747
Infected Population :   206848
Removed Population  :   182042
```


```
Statistics Overview

Days                :    311
Population          :   617594
Heathy Population   :    16102
Exposed Population  :        2
Infected Population :       14
Removed Population  :   601476
```

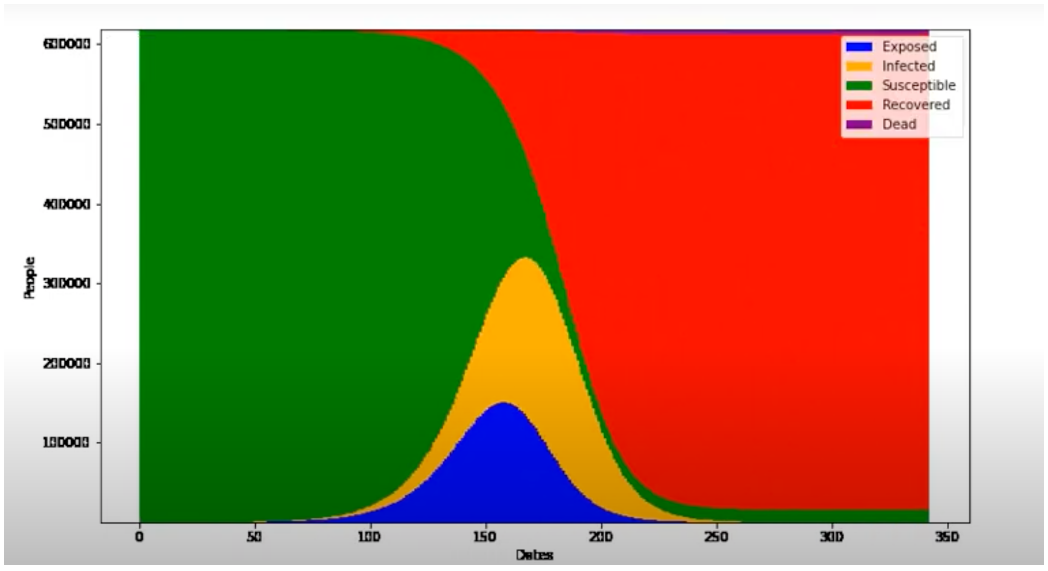
Analysis and Data Visualization

Introduction

After the simulation of transmission, we use python and Jupyter Notebook to create a result visualization for comparisons. We picked up the Exposed, Infected, Susceptible, Recovery, and Dead people number in SEIR model to draw the data visualization.

Design Details

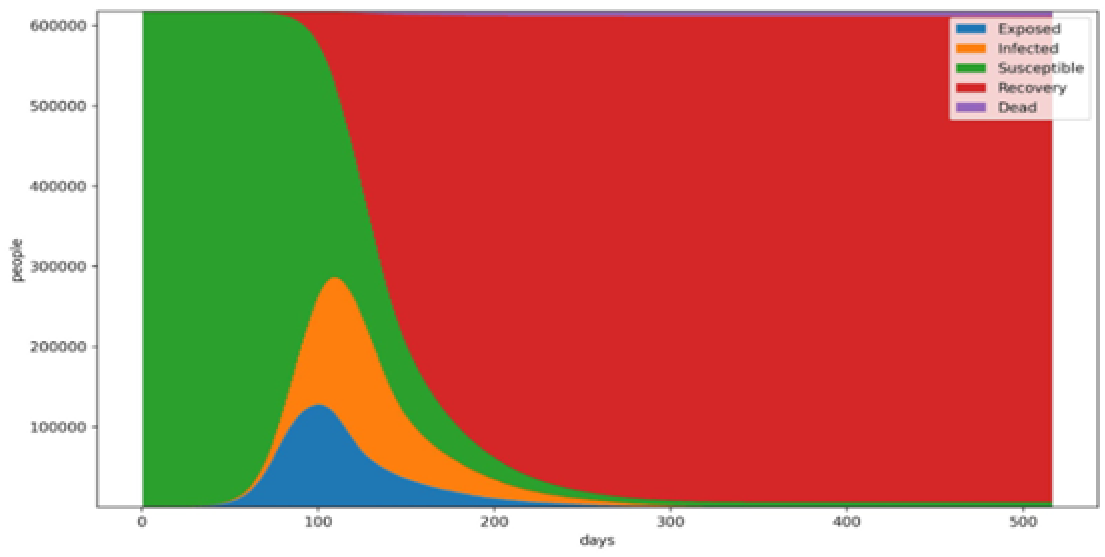
1. Import csv file and clean the data.
2. Extract each parameter as a list to prepare the drawing.
3. Set up x axes and y axes and use matplotlib package to draw.
4. For the animation part, we set up the matplotlib.animation build in function and encapsulate the plot code as a recursion function.
5. Utilized the FFMpegWriter package as a video saver.



Simulation Video

https://youtu.be/3JsQ1emc_T8

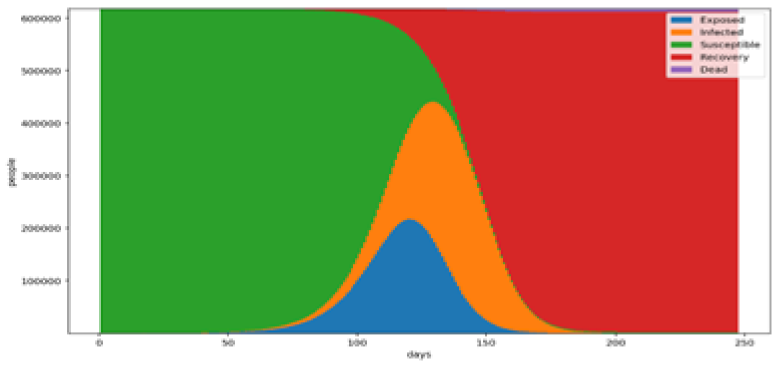
We take this data and do the statistical analysis to implement visual charts.



Taking these parameters into consideration we analysed the data pattern with changing various dynamic parameters.

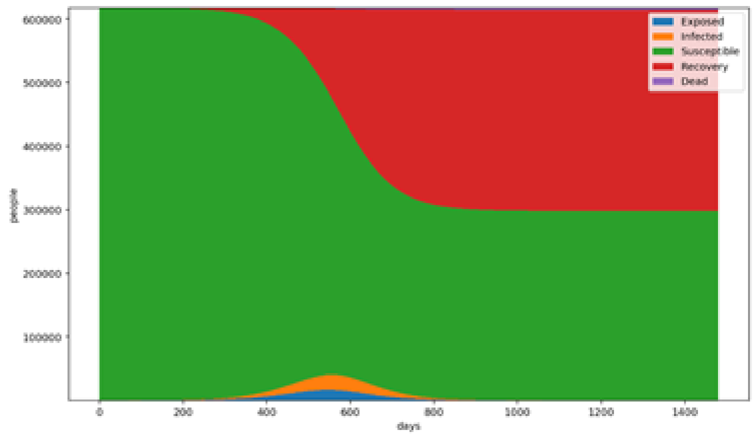
Comparing the mask usage and effectiveness on the virus transmission

1. Transmission of virus without proper mask measures taken



```
Days: 247 Susceptible: 2145          Exposed: 0          Infected: 0          Recovered: 609408          Dead: 6041
```

2. Transmission of virus considering mask measures taken

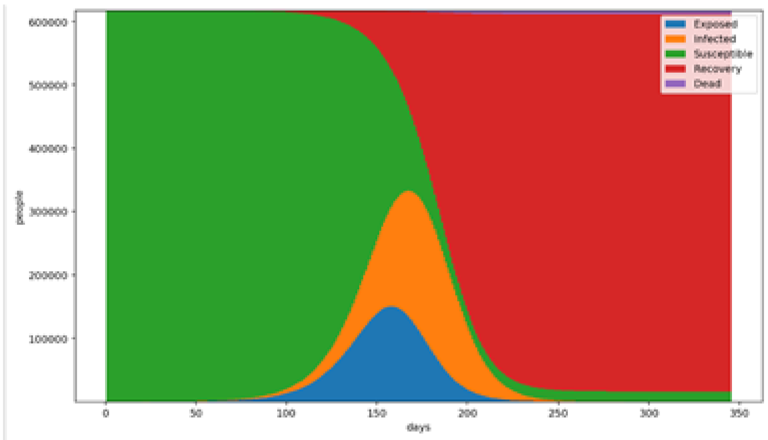


```
Days: 1479 Susceptible: 298341          Exposed: 0              Infected: 0              Recovered: 316001          Dead: 3252
```

Without the proper mask measure, the virus transmission is more and effective, we can see a huge spike in the number of removed and dead people when mask measure is taken light.

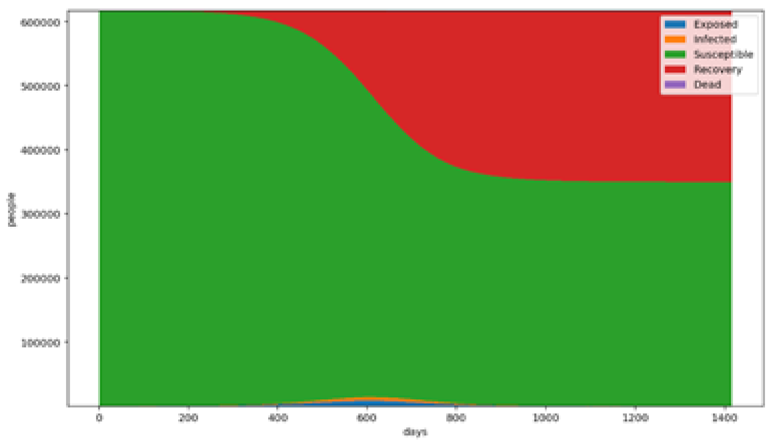
Considering the transmission stats with measures testing and vaccination

1. When the number of tests done per day were low and normal



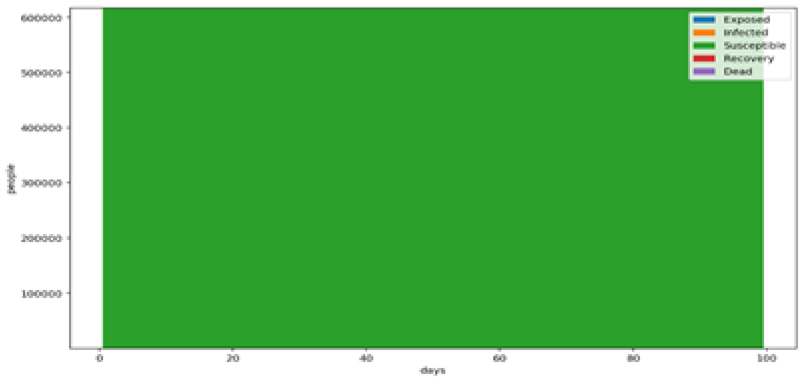
```
Days: 345 Susceptible: 16102          Exposed: 0              Infected: 0              Recovered: 595560          Dead: 5932
```

2. Lets increase the number of tests done per day



```
Days: 1415 Susceptible: 349770          Exposed: 0          Infected: 0          Recovered: 266459          Dead: 1365
```

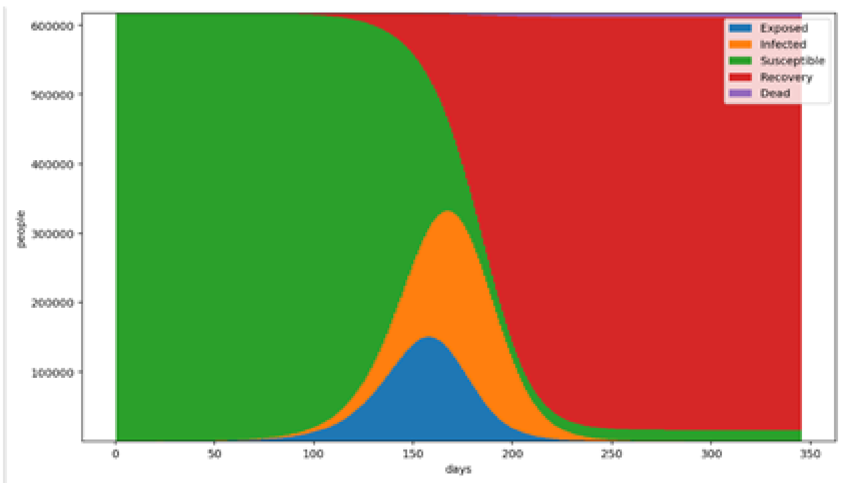
3. With a lot more tests done on the people



```
Days: 99 Susceptible: 617475          Exposed: 0          Infected: 0          Recovered: 119          Dead: 0
```

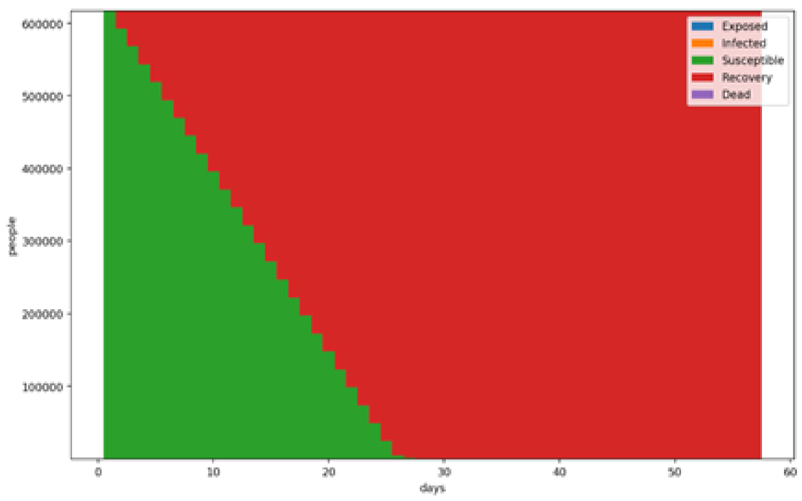
We observed the number of recovered and dead which says that there are more number of infected people when the testing is not done properly. But when we increased the number of tests done per day, the number of removed and dead people gradually decreased.

4. When the number of vaccines per day is normal and low



```
Days: 345 Susceptible: 16102          Exposed: 0          Infected: 0          Recovered: 595560          Dead: 5932
```

5. Let us increase the number of vaccines given per day

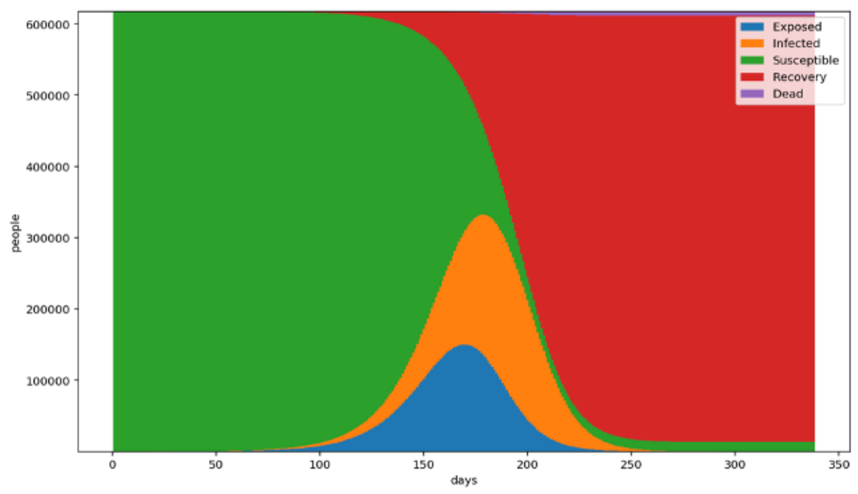


```
Days: 57 Susceptible: 0          Exposed: 0          Infected: 0          Recovered: 617593          Dead: 1
```

When the number of vaccines per day were increased the number of dead were very low compared to the normal situation.

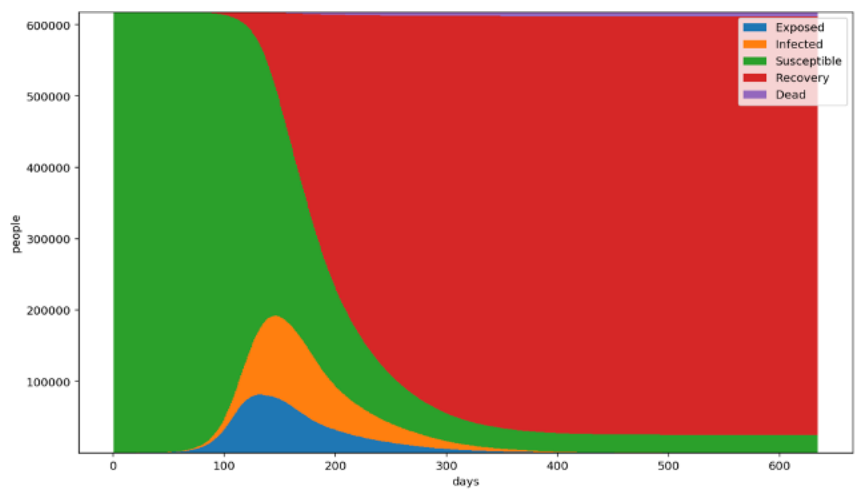
Barrier comparison

When people don’t reduce their activity in the area



```
Days: 338 Susceptible: 13109          Exposed: 0           Infected: 0            Recovered: 598377        Dead: 6108
```

Let’s see what happened if people reduced 90% of their activity area.

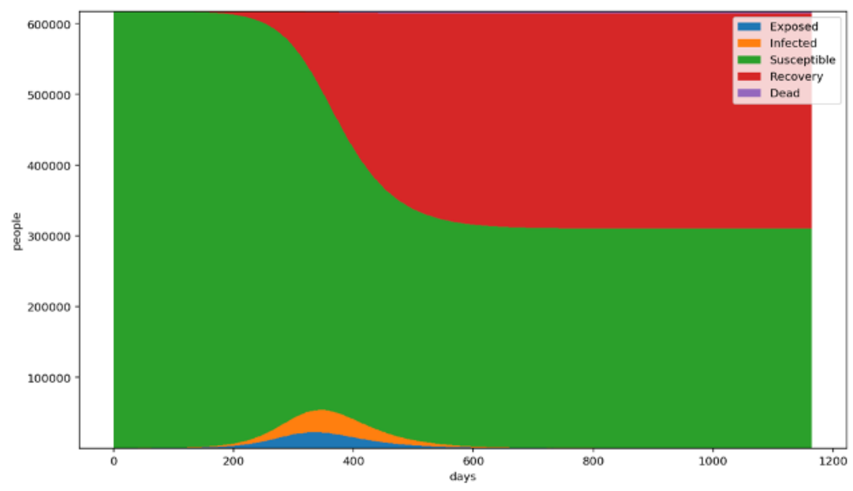


```
Days: 634 Susceptible: 25090          Exposed: 0           Infected: 0            Recovered: 586773        Dead: 5731
```

As we can see, in the scenario of people who reduced their activity area, the infected people number gets much lower and the comparison means that reducing people's activity area is helpful for reducing the spread of COVID-19 virus.

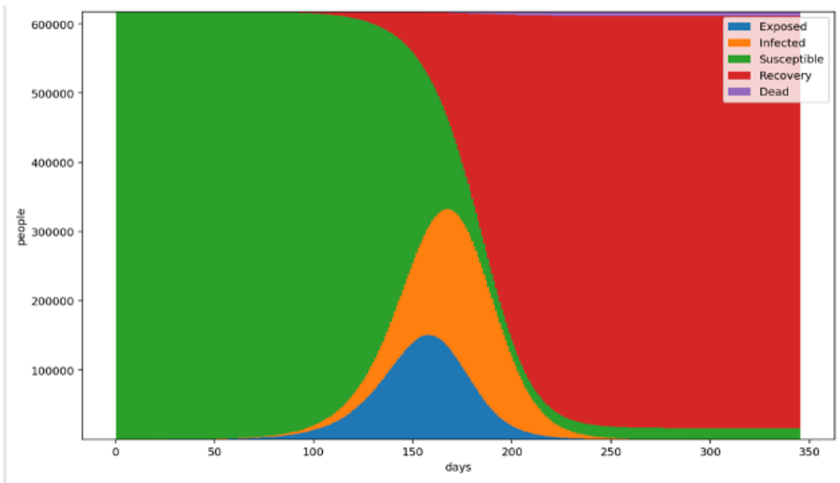
City and Suburban area comparison

The result of covid-19 when it comes to suburban area which the population density is lower



```
Days: 1164 Susceptible: 310432          Exposed: 0            Infected: 0            Recovered: 304058          Dead: 3104
```

Let’s see what happened in city area:

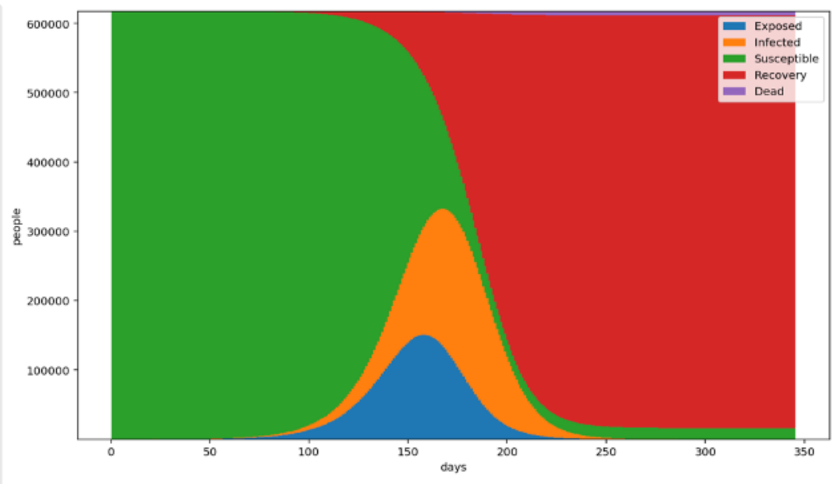


```
Days: 345 Susceptible: 16102          Exposed: 0            Infected: 0            Recovered: 595560          Dead: 5932
```

As we can see, the covid-19 virus spread wider in the city area when compared with suburban areas. Therefore, population density is a quite important factor in virus spreading.

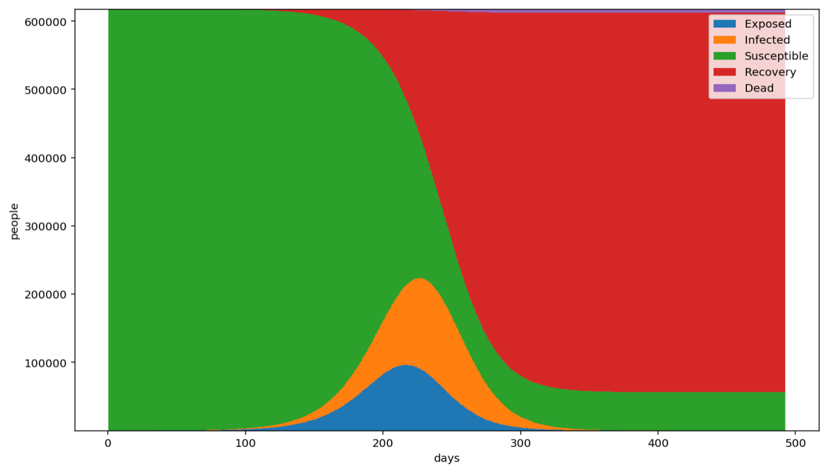
COVID-19 and SARS comparison

The first picture is COVID-19 spreading result with k = 0.1, r = 3.8



```
Days: 345 Susceptible: 16102          Exposed: 0            Infected: 0             Recovered: 595560         Dead: 5932
```

And The second picture is SARS spreading result with k = 0.16, r = 5:



As we can see, the SARS having greater K - factor and R - factor persisted in humans for a long time. Sars spreaded more time and infected less people. On the other hand, Covid-19 is more aggressive.

Conclusion

It is essential to take adequate measures to mitigate the risk of covid-19 virus diffusion. As most of the covid aspects are yet to be discovered, this simulation acts as an alarm to consider precautionary measures on high priority.

With the observations stated above, we conclude that with the usage of precautionary measures like mask usage, testing, and vaccines, the persistence of the covid - 19 virus with humans can be decreased effectively. As the number of tests increased, there was a huge decline in the number of people deceased. As the usage of masks being taken more seriously, we see that the number of people infected with the virus reduces.

Vaccine plays a crucial role in the downfall of the virus transmission. As the number of vaccines per day increased, the number of people declined rapidly.

Reference

1. SEIR Modeling of the Italian Epidemic of SARS-CoV-2 Using Computational Swarm Intelligence (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7277829)
2. Python animation (https://matplotlib.org/stable/api/_as_gen/matplotlib.animation.FuncAnimation.html)
3. FFMpegWriter implementation (https://matplotlib.org/stable/api/_as_gen/matplotlib.animation.FFMpegWriter.html)