Simulation of pandemic behavior in a community of n individuals using Python.

Autor:

Juan E. Gamba A

Asignature:

Artificial intelligence in data driven analysis.

Colombian School of Engineering Julio Garavito

Master's in informatics.

Bogotá D.C

February 1st, 2024

Context

In this document, you will find an analysis of the pandemic model simulation spreading throughout a community with (N) individuals, starting with one infected individual. This community is characterized by having a percentage of its inhabitants attending university (Symbolized as a red square in the center of the screen, see Figure 1).

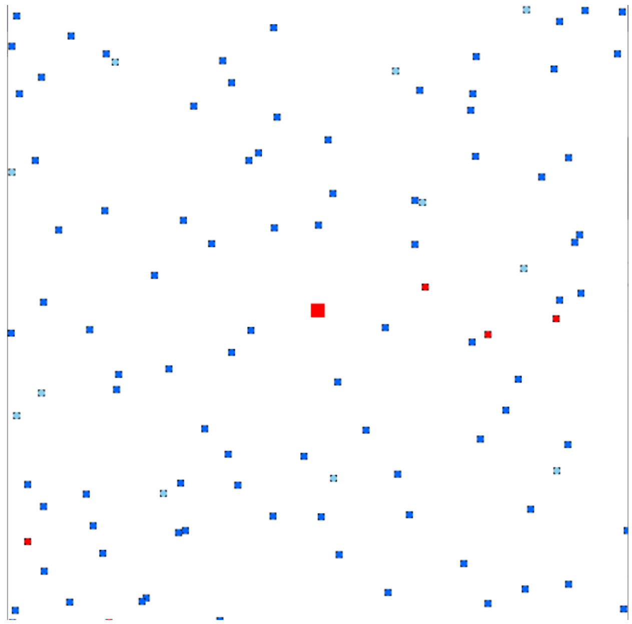

Figura 1.

These individuals who do attend university are randomly generated and are represented with a light blue color. By default, students, faculty, or university staff represent 10% of the entire community, but this percentage can change to observe its impact on infection, death, and recovery curves. Additionally, these individuals move radially between the outer boundaries and the university. Uninfected but susceptible inhabitants are represented with dark blue color, infected ones with red, and recovered ones with green color.

The position of individuals who do not attend university is randomly generated, always ensuring they are outside the perimeter of the university and within the outer boundaries. When an individual reaches an outer boundary, they will automatically appear at the opposite boundary while maintaining the same direction. In each program cycle, the speed and direction of the individual take on a different value. This value can range from [-1,1].

Initial Considerations:

This epidemic model is based on the following initial considerations. I emphasize that they are initial because some of them may change for simulation purposes to observe their impact on the development of the epidemic in the community.

Simulation Time Period

The model will run for a total of 2000 cycles. For this exercise, one day will be equivalent to 66 cycles, resulting in an approximate total of 30 days of simulation.

Recovery Time

This refers to the period of time that elapses from when a person contracts the disease until they fully recover from it. It is important to clarify that the recovery time may vary depending on the individual's immune system response, the existence and effectiveness of treatment, as well as the individual's health status prior to contracting the virus and environmental factors. For the purposes of this exercise, a recovery period of 460 cycles is considered, which is approximately equivalent to 7 days.

Mortality Rate

This can be defined as the likelihood that an individual infected with the disease will die from it [1]. For this exercise, a death probability of 20% will be assumed for individuals between 25 and 50 years old, 35% for individuals between 51 and 80 years old, and 10% for young people between 10 and 24 years old. There will be no individuals under 10 years old or over 80 years old in the community. Ages will be randomly generated following a normal distribution with a standard deviation of 10 years for each age group. (The implementation is shown in Figures 2 and 3.)

Reinfection Probability

Reinfection occurs when an individual recovers after having contracted the virus and becomes infected again with the same pathogen at a later date [2]. In this pandemic model, the probability of reinfection/recurrence is 0%.

```python
def age_distribuiton_gen():
    adults = np.random.normal(loc = 35,scale= 10, size= int(0.6 * self.susceptible))
    seniors = np.random.normal(loc= 65,scale= 15, size = int(0.2 * self.susceptible))
    teenagers = np.random.normal(loc= 18,scale= 6,size =int(0.2 * self.susceptible))
    all_ages = np.concatenate((adults, seniors, teenagers))
    np.random.shuffle(all_ages)
    return all_ages
```

Fig 2.

```python
if 25<=self.age<=50:
    if self.mortality_rate[1] > chance:
        self.kill() #remove the human from infected and susceptibles containers.
    else:
        self.recovered = True
elif 51<=self.age<=80:
    if self.mortality_rate[2] > chance:
        self.kill() #remove the human from infected and susceptibles containers.
    else:
        self.recovered = True
elif 10<=self.age<=24:
    if self.mortality_rate[0] > chance:
        self.kill() #remove the human from infected and susceptibles containers.
    else:
        self.recovered = True
```

Fig 3.

Simulation

The simulation at a certain point in time looks like this.

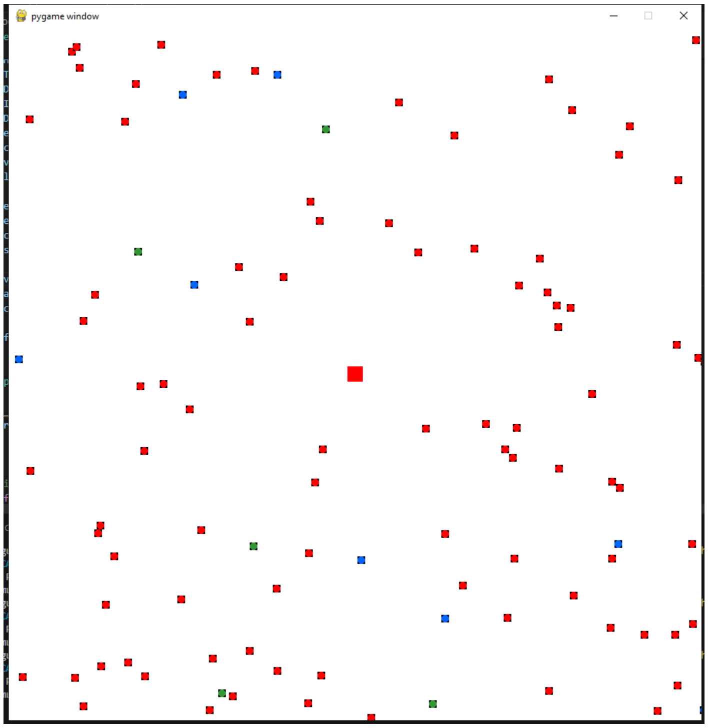

Fig 4.

Most individuals are infected, a few are still susceptible to infection, and others have already recovered. Unfortunately, for this simulation, the peak of infection was 100 inhabitants, which corresponds to 99% of the initial population. The peak of infection occurs on day 9. For that day, there is also one recovered individual and zero deaths (See Figure 5)

Fig 5.

According to the typical SIR (Susceptibles, Infected, Recovered) mathematical model for disease spread, the curve of infected individuals appears to exhibit a reasonable behavior, as this model suggests exponential growth (See Figure 6).

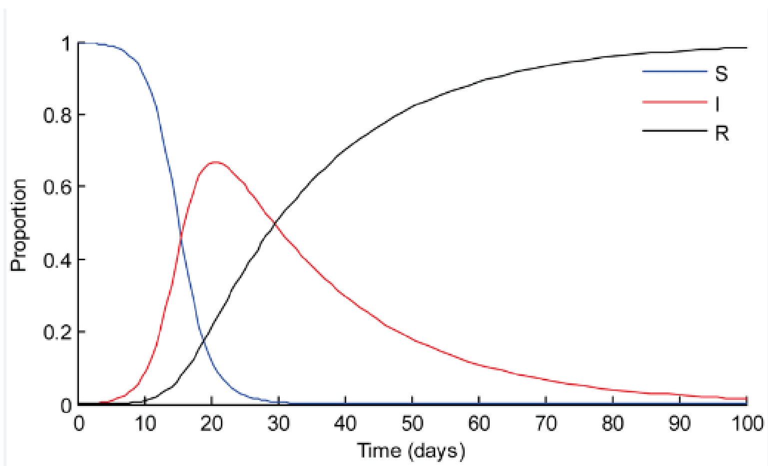

Figura 6. [3]

As the days pass, the epidemic stabilizes and yields the following data (see Figure 7).

Fig 7.

Approximately, the pandemic stabilizes on day 16, with a total of 81 recovered individuals, 14 deaths, and 6 active infections. Similarly, it can be observed that the curve of recoveries behaves very similarly to what the SIR model proposes (See Figure 6).

At the end of the simulation, on day 30, the number of deceased had increased to 18, and the number of recoveries had decreased to 79

Interventions

Quarantine

To simulate a quarantine scenario, infected individuals will enter a resting position at the location where they contracted the virus. Compared to reality, this is a less realistic scenario, as the infected person would likely undergo quarantine at home. However, for the purposes of this simulation, the situation will be simplified. The following figure illustrates how quarantine looks on the 15th day of simulation.

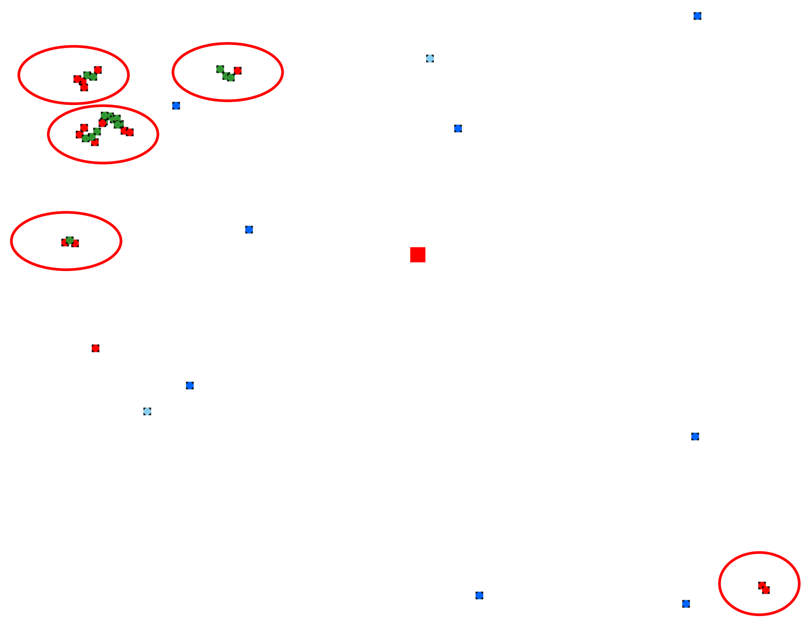

Fig 8.

The peak of infection is significantly reduced from 101 infected individuals to a total of 64.

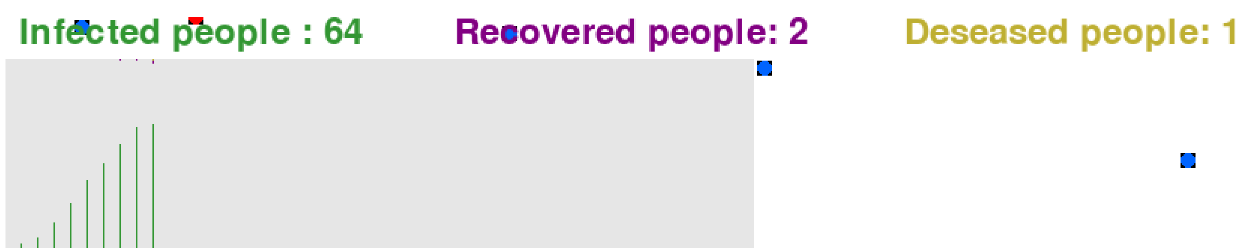

Fig 9.

In the end, a much flatter infection curve can be observed, thereby reducing the negative impact of the pandemic on the community. Applying this scenario to reality, this translates to less overwhelmed hospital centers at the peak of infection and, likewise, reducing recovery times and the number of deaths. By implementing the quarantine measure, the number of deceased individuals decreased to 11.

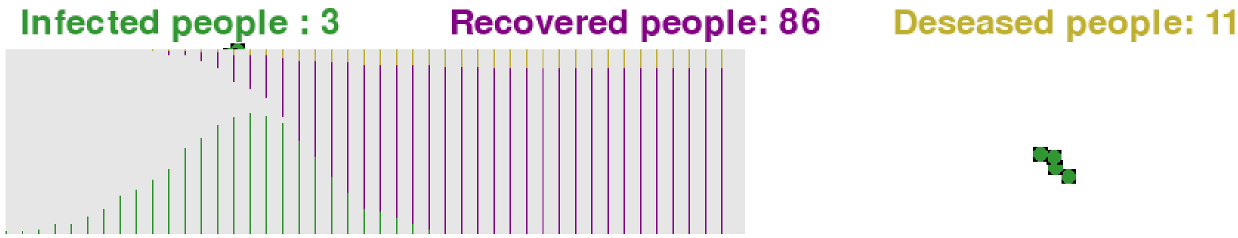

Fig 10.

References

[1] OpenAI (2024). Response to the user regarding the definition of the probability of death in a pandemic.

[2] OpenAI (2024). Response to the user regarding the definition of the concept of reinfection in a pandemic.

[3] https://www.researchgate.net/profile/Claudio-Struchiner-2/publication/47676805/figure/fig2/AS:343729496969224@1458962906357/SIR-model-Schematic-representation-differential-equations-and-plot-for-the-basic-SIR.png

Code References:

[1] Greg Winther (2021) Disease Simulation with Pygame [1/3] [Video]. YouTube. https://www.youtube.com/watch?v=84njPYepKIU&t=1s

[2] Greg Winther (2021) Disease Simulation with Pygame [2/3] [Video]. YouTube. https://www.youtube.com/watch?v=z1NcNodsYTM

[3] Greg Winther (2021) Disease Simulation with Pygame [3/3] [Video]. YouTube. https://www.youtube.com/watch?v=uWrg7Doy_4k