# Classroom Flu Spread Simulation

## Georgia Tech - ISYE 6644 - Simulation | Parth Patel | 04/23/2024

## Abstract

This project presents a comprehensive analysis of a flu outbreak simulation in a classroom setting with 31 students where only one student is initially sick. The simulation explores the dynamics of disease transmission and investigates the impact of various factors on the spread of the virus. The results show that the expected number of infected children aligns closely with theoretical calculations, validating the accuracy of the simulation. The analysis of pandemic lengths reveals that short-lived outbreaks lasting only 3 days are very common due to a low probability of infection and a limited pool of susceptible individuals. Longer pandemics occur within a duration of 13 to 18 days, with rare instances of lengthier outbreaks. Arena Simulation software is employed to fit probability distributions to the generated histograms of pandemic lengths from simulation runs. Additionally, a simulation run is conducted with half of the children being immune at the start by a 50-50 chance, resulting in a significant reduction in the average number of infected children per day compared to previous simulations. These findings highlight the critical role of immunization in mitigating the spread of a virus.

## Background

The Classroom Flu Spread Simulation is a trivial problem that aims to understand disease transmission under some simple circumstances. The problem focuses on a classroom setting in an elementary school that has thirty-one healthy and noninfectious students with the exception for one student named Tommy, who has recently contracted the flu. The simulation begins on day 1 when Tommy enters the classroom and begins interacting with other students. The primary objective is to address key questions such as determining the average number of students infected each day throughout the duration of the epidemic and how long the epidemic lasts. Additionally, the simulation examines the scenario where half of the children are already immune to the flu at the start.

To address these questions, a few assumptions had to be made. It was assumed that every student attends school daily, regardless of their health status. Once a student becomes infected, they remain infectious for a consecutive period of three days following the initial infection. For example, if Tommy infects Rachel on day 1, Rachel will remain infectious on days 2, 3, and 4 of the simulation. Additionally, once a student had been infected for full three days, they were considered immune to the flu, rendering

them incapable of spreading the virus to susceptible students or becoming infected again. For every day of the simulation, I also assumed that every student interacted with every other student in the classroom. The probability of Tommy infecting any susceptible student on any of the three days is 2%. This probability of a successful infection remained constant and equal for every student in the simulation. By assuming independence among all students and days, I was able to model the simulation as a series of independent and identically distributed Bernoulli trials. With these considerations in mind, I used Python to simulate classroom interactions among the students. I conducted a substantial number of replications of the simulation to provide concise and insightful results, offering potential ways into mitigating the spread of the flu within school environments.

# Distribution and Expected Number of Individuals Infected on Day 1

In order to build up and simulate the spread of the flu virus in a classroom, it was first important to understand the initial stages of the simulation. Specifically, I wanted to focus on the first day of the simulation and analyze the distribution of the number of kids infected by Tommy on day one. This distribution can be modeled using a binomial distribution since each interaction between Tommy and another kid is an independent and identically distributed Bernoulli trial with a success probability denoted as $p$. The binomial distribution formula can be expressed as,

$$Pr(X = k) = \binom{n}{k} p^k q^{n-k}$$

where $k$ is the number of kids infected by Tommy on Day 1, $n$ is the number of susceptible kids, $p$ is the probability of infection in a single interaction, and $q$ is equal to 1 - $p$. This leads to the question of determining the average number of kids infected by Tommy on day one. The expected value of a binomial distribution, which represents its mean, can be calculated using the formula:

$$E(X) = np$$

In this case, the expected value for $n$ = 30 (as there are 30 susceptible kids) and $p$ = 0.02 (the probability of a successful infection in a single interaction) is simply 0.6 as shown below:

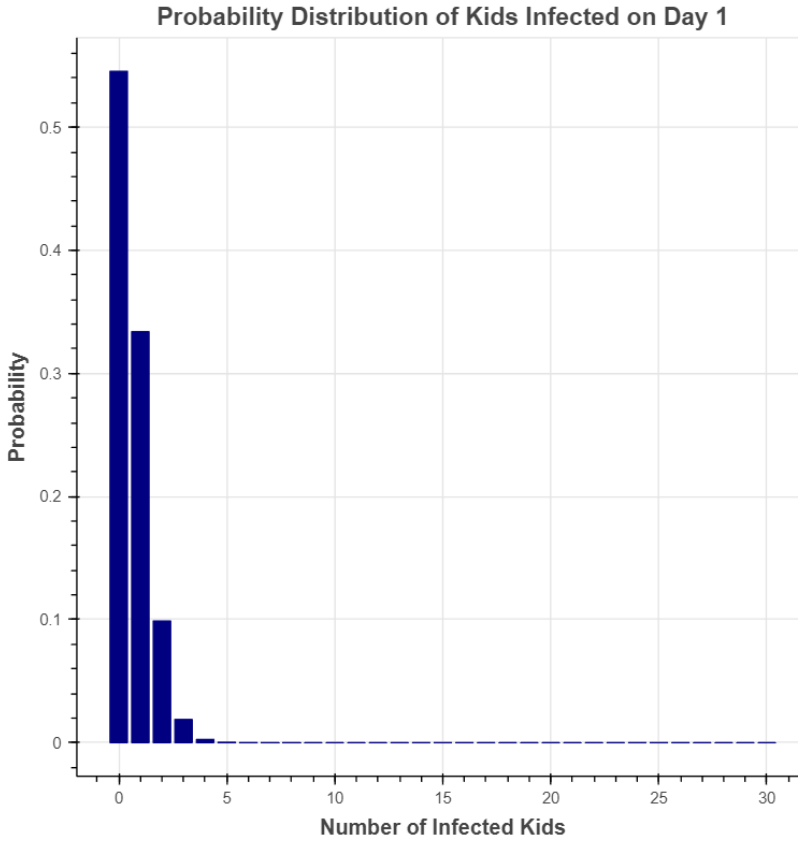
$$E(X) = 30 * 0.02 = 0.6$$

This can be further confirmed using the binomial probability mass function from the SciPy Python library and plotting the probabilities for each $X$ value. Running the following code provides the given output and plot:

```python
import scipy.stats as stats

n = 30  #number of healthy kids, so excluding Tommy
p = 0.02  #probability of infection

X = list(range(n+1))  #values of X (0 to n+1)

pmf = stats.binom.pmf(X, n, p)

for x, prob in zip(X, pmf):
    print(f'P(X = {x}) = {prob:.4f}')
    if (prob < 1/10000):
        break

print("\nExpected value given n =", n, "and p =", p, "is:", stats.binom.expect(args=(n, p)))

import plot_helper as plot
plot.prob_distr(X, pmf)
```

1.3s Python

```
P(X = 0) = 0.5455
P(X = 1) = 0.3340
P(X = 2) = 0.0988
P(X = 3) = 0.0188
P(X = 4) = 0.0026
P(X = 5) = 0.0003
P(X = 6) = 0.0000

Expected value given n = 30 and p = 0.02 is: 0.6
```



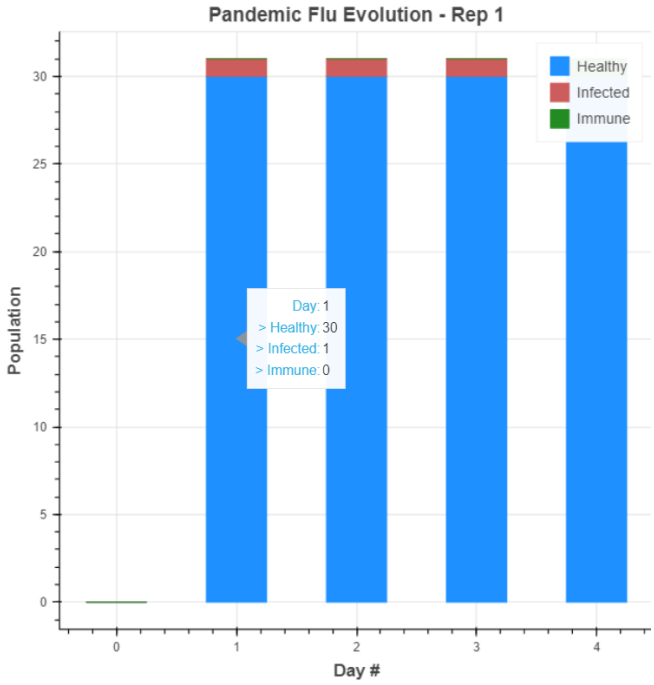

Therefore, the expected number of kids that Tommy infects on day one is 0.6, which means that on average, we would expect Tommy to infect approximately 0 or 1 kid. However, when I looked at the probabilities more closely, there is a higher chance that Tommy will not infect anyone. The probability of Tommy infecting zero kids on day one is 0.5455, which is higher compared to the probability of infecting at least one kid, which is 0.334. Therefore, based on these probabilities, it is more likely that Tommy will not infect anyone on day one.

## Forming the Foundational Code and Functions

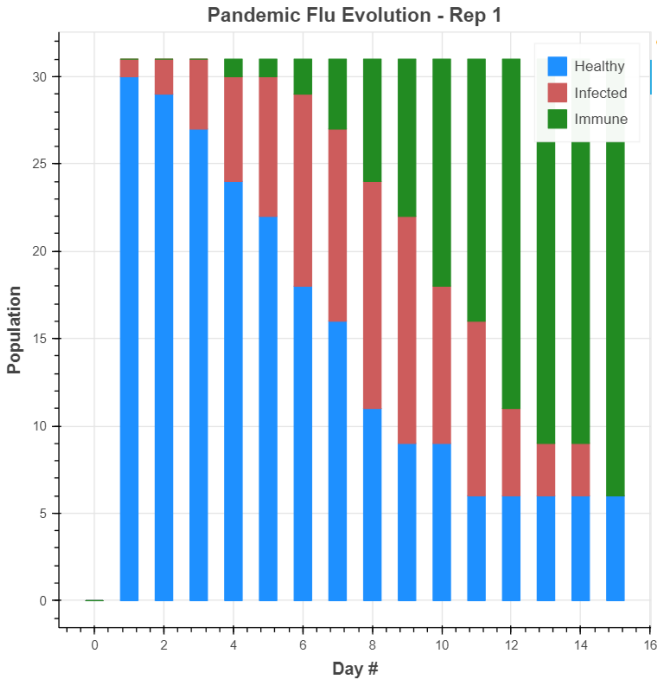
Based on these results, there was a need to further investigate the expected number of children infected by day two, including Tommy. Moreover, it is also important to analyze the daily number of infected children throughout the entire epidemic. To do this, a simulation of each day of the epidemic can be carried out using Python. This involved coding a mechanism where all children interacted with one another and had an equal probability of spreading the flu. By monitoring the number of infected, healthy, and immune children each day, it became possible to determine the duration of the epidemic. The following three Python functions were developed and stored in the "sim_funcs.py" file that is enclosed within the project folder.

The function `run_day_in_rep` was designed to simulate a single day within the epidemic scenario. It takes several parameters such as the number of kids, the probability of infection, the duration of recovery, a Pandas data frame, and a debug print option. The function begins by setting a random seed for the day to ensure reproducibility. It then identifies the indices of currently infected kids from the data frame and tracks their infection duration. It begins simulating the spread of infection by allowing each infected child to interact with healthy children, determining whether an infection occurs based on the infection probability. Once the specified recovery period has been met, the infected children are marked as immune and their infection status in the Pandas data frame is updated. The function updates the data frame accordingly and returns the updated data frame to finish the day. This function is key to the simulation and keeps a daily log of the classroom and the health status of each child.

Another function called `run_rep` conducts a replication of the entire epidemic scenario. This is important since frequently, Tommy will not infect anyone and the epidemic can end in just three days. Other times, the epidemic may last much longer and this length can vary greatly over many replications so it is important to run many iterations of the simulation to get accurate results. This function also takes in parameters such as the number of kids, the probability of infection, the recovery period, a Pandas data frame, and options for printing debug information and plotting each day's results. The function starts the simulation by setting the day count to one and creating a copy of the initial data frame. This initial data frame just stores information about every student with the special case of Tommy who is the only one infected at the start of the simulation. It then iterates through each day while at least one child is infected. For each day, the function executes the `run_day_in_rep` function mentioned above to update the data frame, track the number of infected and immune children, and create a report with the daily results. The function continues until no child is infected, and if specified, it can plot the evolution of the pandemic for each day. In the end, it returns the report containing the cumulative results of the epidemic simulation.




*An example of a pandemic evolution plot where Tommy does not infect anyone and the epidemic ends in just 3 days.*

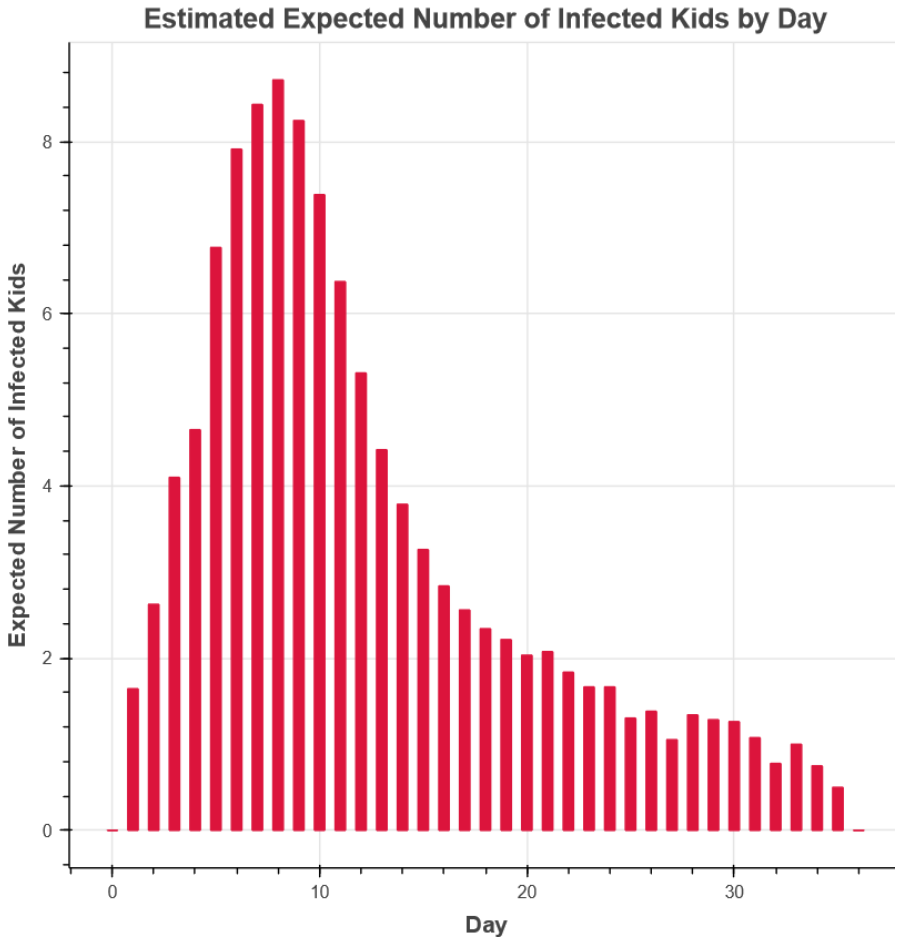



*An example of a pandemic evolution plot where Tommy infects other students and the infection spreads throughout the classroom until either everyone is immune or when the flu is no longer spreading to healthy children.*

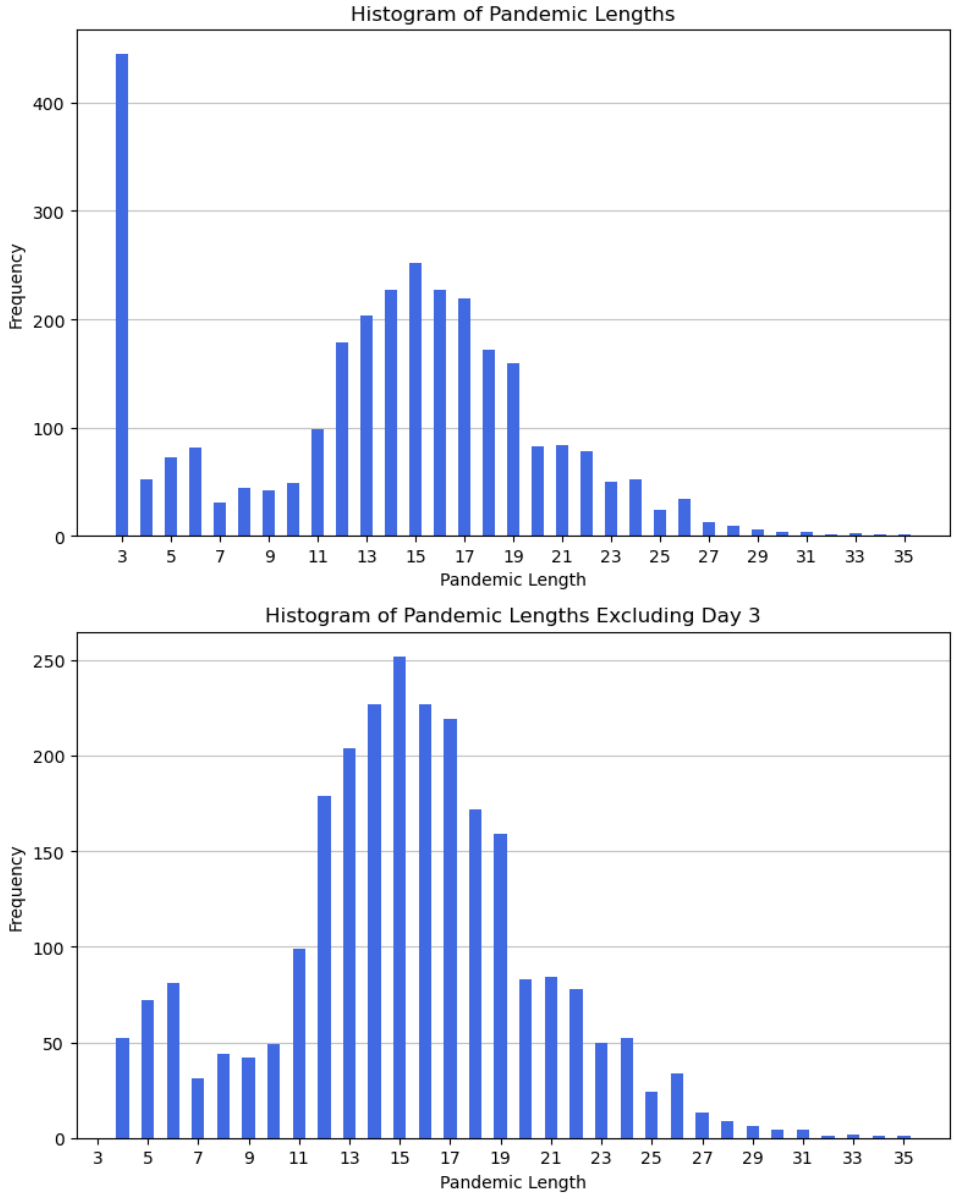
Finally, the function run_sim executes a complete simulation of the pandemic flu spread. It takes parameters such as the number of kids, the probability of infection, the recovery period, the number of repetitions, options for printing debug information, and plotting each day's results if the number of reps is less than five. It iterates through the specified number of repetitions, creating a Pandas data frame with only one initial infected kid at the start of every replication and executing the run_rep function to simulate the epidemic for each replication. The simulation results for each repetition are stored in a dictionary which is then returned by the function. The results can be used to visualize the expected number of infected kids for a given day as well as to plot a histogram of the epidemic lengths.

## Simulating an Epidemic with a Single Infected Child

When running the three functions together in the provided Jupyter notebook with a substantial number of replications, valuable insights can be gained into the dynamics of the simulated flu outbreak. For this project, I conducted 3000 replications with simulation parameters of 31 kids, a probability of infection of 0.02, and a recovery period of 3 days. The results from this simulation run are presented below.

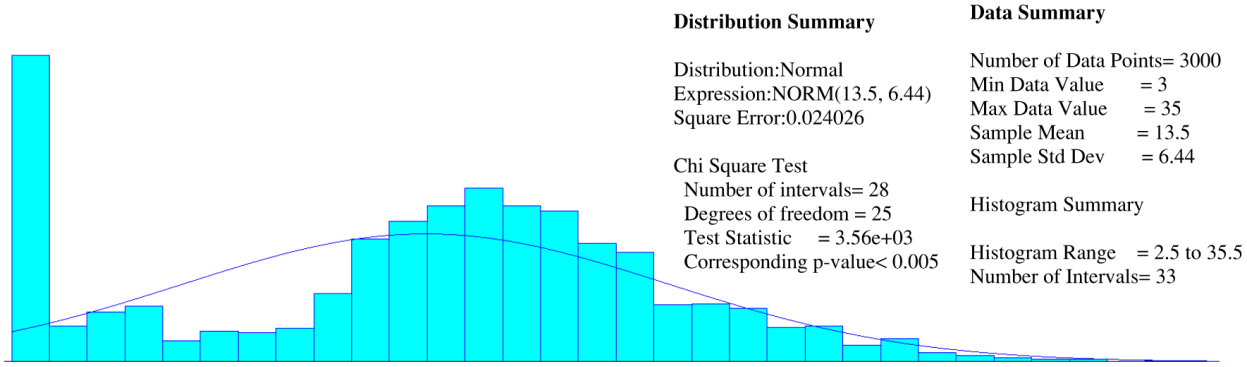


By examining the graph of the expected number of infected kids over time, I observed a rapid increase in the number of infected children during the initial days, followed by a gradual decline as the pandemic progressed and more kids developed immunity to the flu. According to the graph, the highest number of infected kids averaged 8.717, or 9 kids, and occurred on day 8 of the simulation. This average was derived from 3000 replications and suggests that the majority of children become infected within the first few days of the epidemic. Around day 20, the average number of daily infections drops to only 1 or 2 kids per day. Notably, the expected number of infected kids on day 1 aligns precisely with the earlier discussion, where I calculated the expected number of kids Tommy would infect on the first day. This number was 0.6 and meant that, on average, we expected Tommy to infect approximately 0 or 1 kid on day one. The graph above provides the result of an average 1.645 infected kids on day one and this falls within the expected range of 1 to 2 infected kids when counting Tommy in the total number of infected kids.

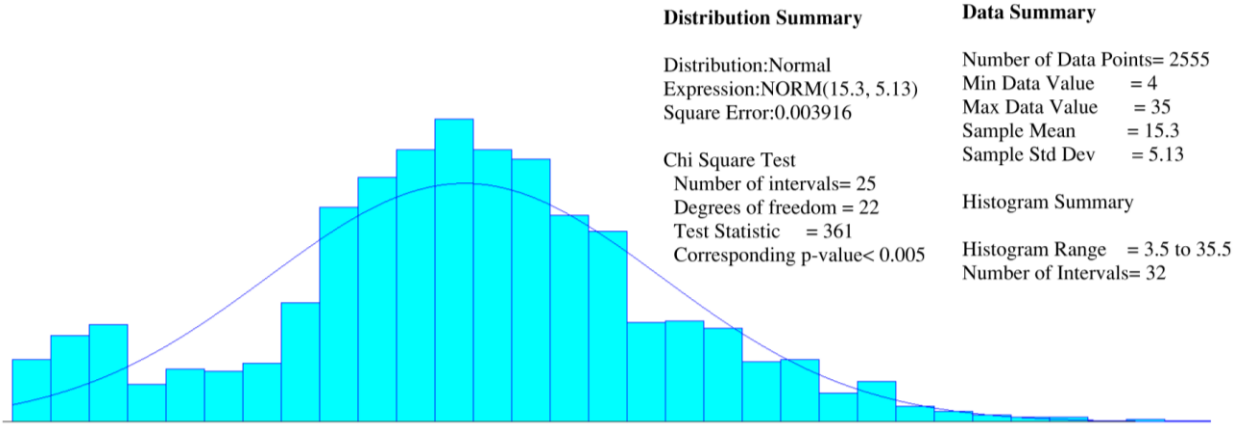


Additionally, the simulation provided histograms depicting the distribution of pandemic lengths across all replications as seen above. The reveals that in 445 replications out of the 3000 total, or 14.83% of the time, the pandemic concluded within just 3 days. This outcome can be attributed to the low probability of successful infection of 2% and the limited number of susceptible children in the classroom of which there were 30 in total. This means that Tommy failed to infect any individuals during the first three days which resulted in a short-lived epidemic. To gain a deeper understanding of pandemic lengths outside of this special case, I made another histogram excluding pandemics that lasted only 3 days. The second histogram shows that most pandemics lasted between 13 and 17 days, with frequencies ranging from 200 to 250 for each day within this range. While some rare cases extended up to approximately 35 days, such instances occurred infrequently and only a handful of times across the 3000 replications.

Finally, I wanted to determine the best-fit probability distribution for the histograms and I chose to do that using Arena Simulation software by Rockwell Automation. Arena provides an Input Analyzer tool that can fit multiple distributions to some given data and returns the best fit distribution. I supplied Arena with both histograms above and the results are presented below.




Histogram of Pandemic Lengths

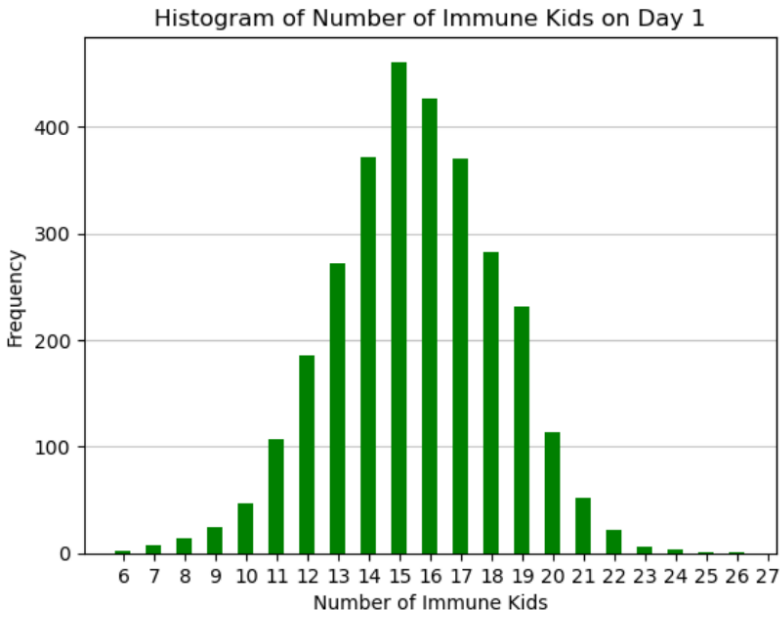



Histogram of Pandemic Lengths Excluding Lengths of 3

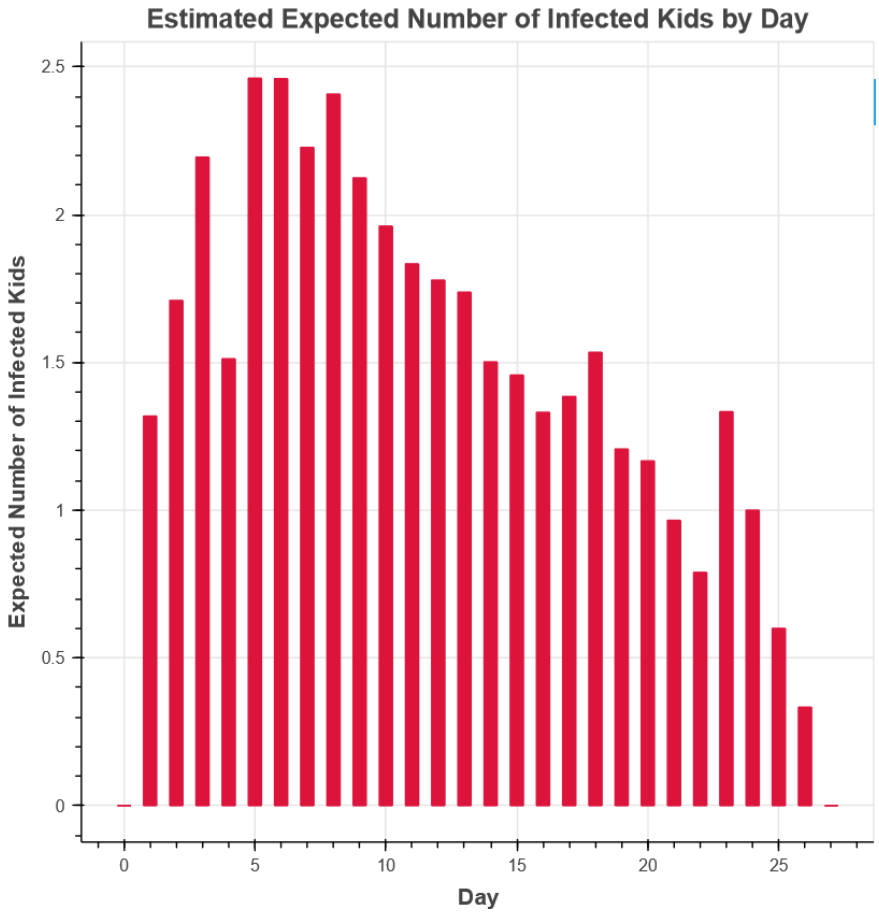
The distribution summary reveals that a normal distribution provided the best fit for both histograms. The histogram encompassing all pandemic lengths had a mean of 13.5 days with a standard deviation of 6.44 days. On the other hand, the histogram excluding lengths equal to 3 had a mean of 15.3 days with a standard deviation of 5.13 days. Notably, the squared error was significantly lower for the second histogram, measuring 0.0039 compared to 0.024 for the first histogram, which indicated the normal distribution as a better fit for the second histogram. However, it is important to note that while the results obtained from Arena Simulation software suggest a normal distribution as the best fit, I am not confident in the accuracy of these results. During my testing of the simulation runs, I frequently observed Arena fitting the gamma and beta distributions as well. If I were to revisit this project in the future, I would explore alternative approaches such as using SciPy in Python or community libraries like Fitter to fit the data and obtain more accurate distribution fitting results.

## Simulating the Impact of 50-50 Immunization on an Epidemic

The other key question I set out to answer was assessing the impact that a 50-50 chance of every child being immune at the start of the simulation had on the results above. To do this, I created another function, `run_sim_half_immune` in the "sim_funcs.py" file that is very similar to the `run_sim` function mentioned earlier. The only key difference is that in this simulation, each child has a 50-50 chance of already being immunized on day one in the initial data frame. Therefore, I expected that the pandemic should last for a shorter duration during each replication and so less kids should be infected overall. For this modified simulation run, I once again conducted 3000 replications with 31 kids, a probability of infection of 0.02, and a recovery period of 3 days. The results obtained from this modified simluation are presented below.

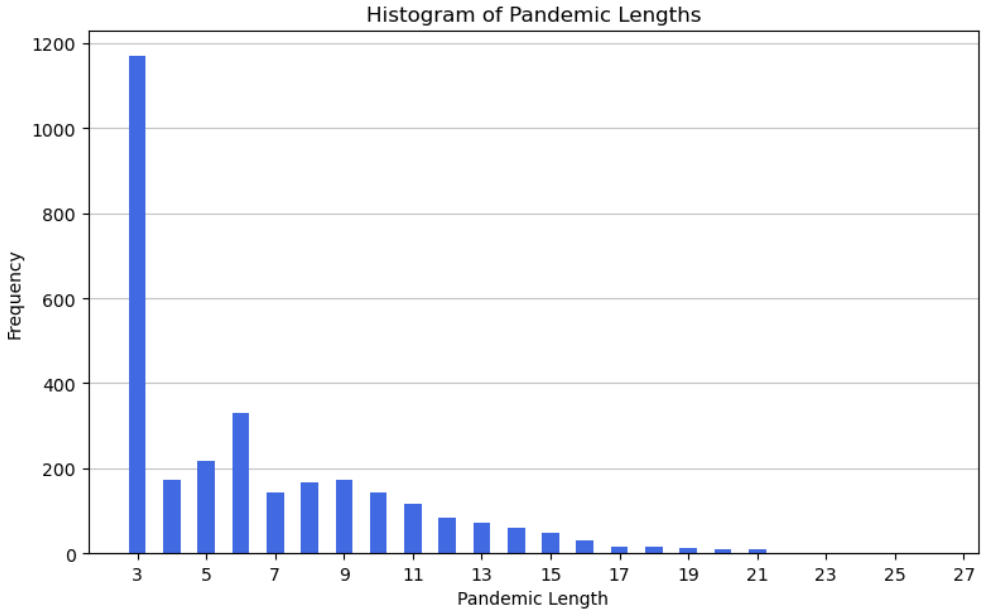


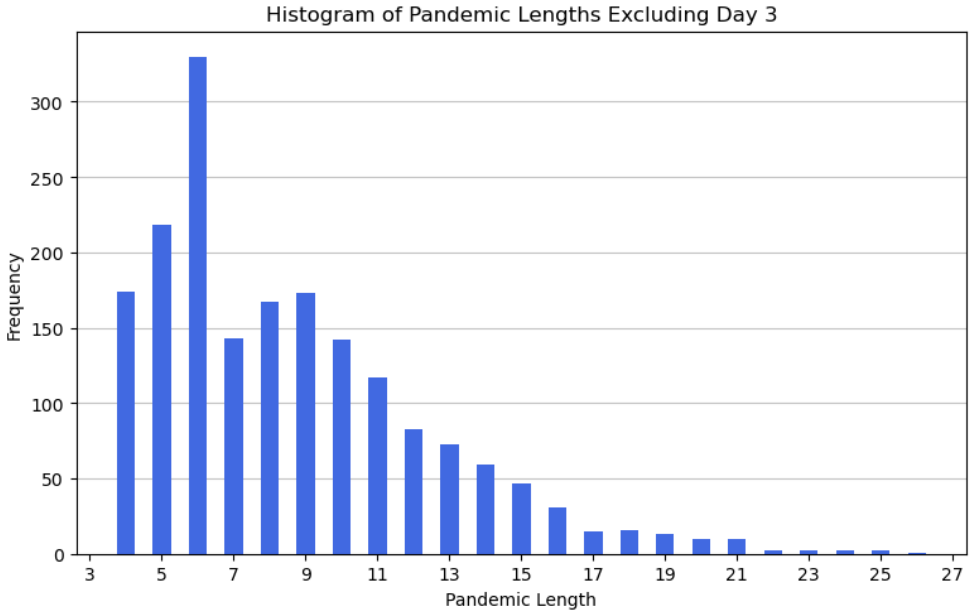
The histogram above shows the number of immune kids at the start of each replication in the simulation. This was simply to verify that the randomization function responsible for determining whether a given child is immune or not was working properly. As evident from the histogram, it approximately divides the kids into two equal groups, exhibiting a distribution that closely resembles a normal distribution. According to the law of large numbers, this distribution should have a mean equal to half of the population. In this case, I observed that most of the time, around 15 to 16 kids out of a total of 30 kids were immune at the start of each replication. This confirmed that the randomization function was working as intended and the rest of the results could be analyzed.




The estimated expected number of infected kids per day plot above shows a significant reduction in the number of infected children compared to the previous simulation run. The highest averages were observed in the initial days of the simulation between days 5 to 8. However, the key difference is that the average number of infected kids ranges from around 2 to 2.46, translating to 2 to 3 kids being infected. As the simulation progresses, this average declines sharply to 1.17 to 0.6 kids between days 20 to 25. By immunizing half of the kids on average at the start of the simulation runs, the flu outbreak reaches fewer

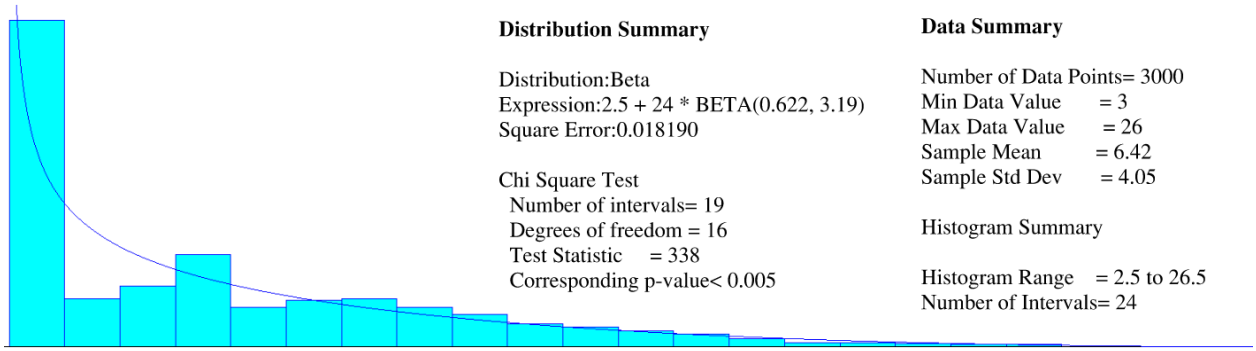
children and the pandemic concludes much more rapidly. This is evident from the histograms provided below.



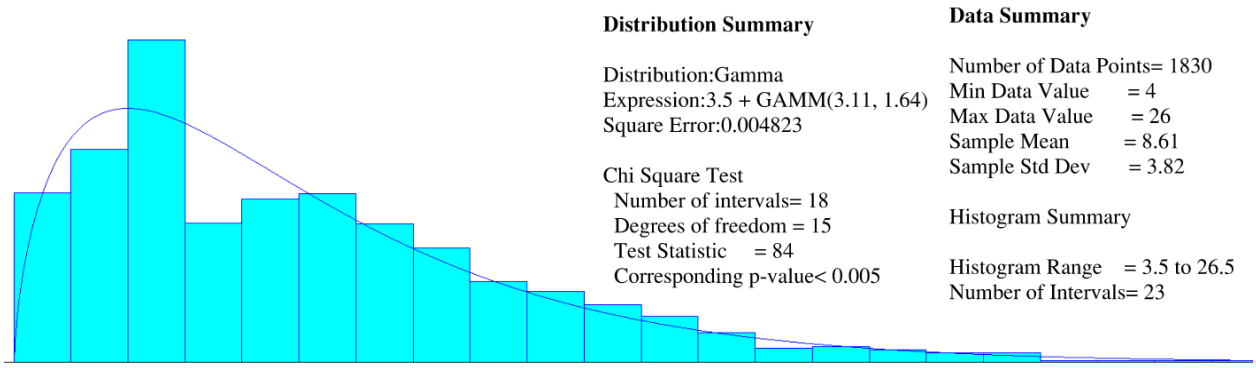




The histograms of the pandemic lengths above reveal that in 1170 out of the total 3000 replications, or 39% of the time, the pandemic ended in just 3 days. This represents a substantial increase from the previous simulation run where no one was immune at the start and only 14.83% of the pandemics lasted 3 days. Moreover, when excluding the 3-day pandemics, it can be observed that the majority of replications had pandemic lengths ranging from 4 to 11 days. Even in the rare cases, the pandemics only reach a maximum of 26 days on a few occasions. These results highlight the significant impact that immunization to a virus has on the spread and control of an outbreak.

As before, I wanted to use Arena to see which probability distribution fit best to each of the histograms. The results are shown below.




Histogram of Pandemic Lengths - Half Immune at Start of Simulation Runs




Histogram of Pandemic Lengths Excluding Lengths of 3 - Half Immune at Start of Simulation Runs

According to the distribution results, the histogram of pandemic lengths including all lengths was best fit by a beta distribution, while the histogram excluding lengths equal to 3 was best fit by a gamma distribution. As mentioned already, I am not confident in the accuracy of these best fit results as they varied between different simulation runs. The notable conclusion drawn from these findings is that, in the majority of cases, pandemics had a relatively short duration when Tommy infected another student within his first three days of being infectious. This duration was considerably shorter than the previous simulation run where no one was immune to the flu at the start.

# Conclusion

The comprehensive analysis of the flu outbreak simulation gave several significant insights. The expected number of infected children aligned closely with the theoretical calculations which validated the accuracy of the simulation in capturing the dynamics of the outbreak. The analysis of the pandemic lengths also revealed a notable pattern where short-lived outbreaks lasting only 3 days were very prevalent, primarily due to the combination of a low probability of infection and a limited pool of susceptible kids.

These small outbreaks indicate the effectiveness of preventive measures and the rapid control of the virus within the simulated classroom environment. The advice to simply stay home when one is sick is proven repeatedly and should be followed whenever possible to prevent the spreading of contagious diseases to a bigger population. Furthermore, the distribution of longer pandemic lengths was mostly between 13 and 18 days which suggested a moderate level of sustained transmission. Occurrences of lengthier outbreaks were rare given the simulation parameters for this project.

In the simulation run where half of the kids were immune at the start by chance, the average number of infected children per day showed a significant reduction compared to the previous simulation. The pandemics in this scenario also had shorter durations with a significant increase in the occurrence of 3-day pandemics. These results emphasize the crucial role that immunization has in controlling the spread of infectious diseases and minimizing their impact.

The findings from this project provide several ideas for future work in this area of simulation. One potential idea is to incorporate more complex disease models that consider factors like varying infectiousness over time, different modes of transmission, or the impact of interventions like school closures, social distancing, vaccination doses, and face masks. Using a more complicated interaction system for the population can also make this simulation align more closely with real word cases. Overall, this project could serve as a solid foundation for further research in understanding and managing disease outbreaks not only in educational settings but also in general populations.

# References


Koehrsen, Will. “Data Visualization with Bokeh in Python, Part I: Getting Started.” *Medium*, 20 Mar. 2018, towardsdatascience.com/data-visualization-with-bokeh-in-python-part-one-getting-started-a11655a467d4.

Mccarthy, Kevin. "Does the flu provide better immunity than a flu shot?" *New York Times*, 28 Oct. 2016, p. D4(L). *Gale Academic OneFile*, link.gale.com/apps/doc/A468531141/AONE?u=anon~3cd33e50&sid=googleScholar&xid=34404ef0. Accessed 21 Apr. 2024.

“Numpy - Fitting Empirical Distribution to Theoretical Ones with Scipy (Python)?” *Stack Overflow*, stackoverflow.com/questions/6620471/fitting-empirical-distribution-to-theoretical-ones-with-scipy-python.

“Numpy.random.choice — NumPy V1.22 Manual.” *Numpy.org*, numpy.org/doc/stable/reference/random/generated/numpy.random.choice.html.

"Pandas.DataFrame — Pandas 1.2.4 Documentation." *Pandas.pydata.org*, 2023, pandas.pydata.org/docs/reference/api/pandas.DataFrame.html.

"Statistical Functions (Scipy.stats) — SciPy V1.3.3 Reference Guide." *Scipy.org*, 2019, docs.scipy.org/doc/scipy/reference/stats.html.