1. Summary

Faced with the challenge analyzing an unknown disease and determining appropriate actions, our team developed a model to determine the behavior of the disease and simulate its spread in the village. To add accuracy to our model, given relatively little information, we ran the model countless times with different parameters in an attempt to generalize the disease's behavior.

We first looked toward the SIR model as the foundation of our simulation. We split the people of the village into types: Susceptible, Infected, Recovered, as well as Dead. We also accounted for the potential latent period of infection. Next we looked at the Rumor Spreading Model for insight. We had assumed the disease was spread through human interaction rather than the use of vectors or other means. Therefore, we compared the spreading of a rumor to the spreading of a disease, as both require human interaction and involve spreaders and receivers. We manipulated the formula of rumor spreading to fit our model of the spreading of infection, and determined the probability of a particular infected person infecting a susceptible person.

From these two models, we coded using Java our own model that simulated the spread of infection. We used the given values of infected and susceptible to start with and allowed the members of the village to interact each day. However, since we did not know many values crucial to determining the behavior of spread, we set them as modifiable variables, seeing what would happen under different assumed conditions. We ran this program until the number of infected people reached 0; in other words, the disease is exterminated.

We analyzed the data we received to classify the type and severity of the disease, and whether or not the epidemic was contained. We concluded that this disease followed the behavior of a point source outbreak, and the epidemic was quickly contained. From further analysis of our data, we also devised recommendations to prevent the spread of this disease: namely, preventing contact with infected individuals as well as preventing contact between this village and any other village.

When more information was learned about the particular disease, we were able to create a more accurate model. We were also able to take into account different situations--including the spread of the disease to other municipalities. However, we have faith that the nature of the disease and the consideration of our recommendations will prevent the continued spread of this dangerous disease. r

EPI-DEALING WITH EPIDEMICS

HIMCM PROBLEM B

TEAM #4637

Table of Contents

2. Introduction

When the infectious Ebola virus reared its ugly head in Africa in the summer of 2014, many people were terrified. How could this virus, previously a seemingly unknown and non-existent threat to humans, cause such devastation in such a short amount of time in many parts of West Africa? In just 6 short months, more than 13,500 cases have been reported, and 4951 humans have perished from this deadly disease. This catastrophe not only devastated and terrified the public, but just as importantly, it also warned them about the necessity of proper disease control methods to prevent a similar tragedy from repeating itself. In addition, it demonstrated the potential damage that a scarcity of proper medical resources, such as doctors, containment centers, money, and drugs, can cause in developing countries, where medical care is not ubiquitous. Therefore, when a strange, unknown disease began to pop up in an isolated village in Indonesia, an ominous parallel between this potential threat and that in West Africa seemed strikingly clear. In this problem, we are tasked with analyzing this potential epidemic in Indonesia, creating a mathematical model to understand the behavior of the disease, as well as determining how to contain it to the best of our ability. Much like the events in West Africa, we are faced with the challenge of limited resources and what ostensibly is a very deadly disease. However, in our case, we will do whatever we can to ensure the tragedy of Ebola does not repeat itself.

What is the background of this novel disease in Indonesia? It was first discovered by a humanitarian mission to the village in question. When they arrived, they found out that 15 people infected by the disease had died in the week before they came, leaving 300 in the village. Slightly less than half of the village’s current inhabitants are showing symptoms of this disease, while the rest--slightly more than half--are not. It is possible that the disease in its current form is a mutation of a similar disease that afflicts a native Indonesian animal. Many similar diseases, which just popped up and previously unknown to be infectious to humans, descend from an animal counterpart. For example, a type of chimpanzee in West Africa was the original source of HIV infection in humans; the counterpart SIV (Simian Immunodeficiency Virus) was before known to affect chimpanzees only. In addition, the previously mentioned Ebola virus is also known to have spread to humans from a monkey infected with a similar disease.

We are faced with the challenge of controlling this unfamiliar disease. To do so, we will first make a mathematical model simulating the disease spread within the village in order to gain a better understanding of this disease, including its potential spread, infectiousness, and classification of spread. In addition, we will determine whether or not this disease can be classified as a contained epidemic. According to WebMD, an epidemic is defined as when an infectious disease spreads rapidly to many people. An epidemic is uncontained when the number of infectious people is increasing, and decreasing when it's not. Next, we face the task of allocating scarce resources to combat and control the disease, including preventing it from spreading and treating already infected inhabitants. We have limited amounts of resources, including doctors, containment facilities, money, research, and serums, to combat this disease, which is now being monitored by our employer, the International World Health Organization. Our goal is to develop a model that accomplishes all of these tasks.

From the findings of our model, as well as research and observations, we will need to give recommendations on the best course of action to national Centers of Disease Control in order to prevent the spread of this disease. At some point, we expect to receive information from an international research team that will be studying this disease first hand, which we can use to further improve our model. Based on their direct observations, they will be able to provide us with more specific information about the behavior of the disease. In addition, it is possible that that they may be able to give us more information upon our request. After completing our research, we will debrief news outlets around the world on the state of crucial matter, and educate the public about the best practices to help stem the spread of this disease.

3. Assumptions with Justifications

Assumption 1: All symptoms are caused by one disease. An individual who is showing these symptoms has the disease.

***Justification*:** The situation only mentioned that many inhabitants are showing similar symptoms. We will assume that only one disease is causing these similar symptoms, rather than a combination of two or more diseases, as the chances of this possibility are extremely low, and no information was given in the problem that would justify this. This assumption is necessary to build a feasible model.

We will also assume that any individual who shows these symptoms has the disease. As we mentioned earlier, no other disease would cause these symptoms. In addition, the problem implies that the symptoms of the disease are noticeable, an uninfected individual will also not display these symptoms. Thus, an individual will have to be infected in order to display these symptoms.

Assumption 2: The disease in question is deadly/scary enough that any logical person will avoid being infected by taking preventative precautionary measures.

***Justification*:** Due to the fact that almost half of the village is infected, and 15 have died, it is reasonable to assume that people regard this disease with apprehension. In addition, since the International World Health Organization and major authorities are involved, word about the danger of this disease will have spread to neighboring villages and islands. Due to this, uninfected people will have less interaction with infected people showing symptoms. In addition, it will be less likely that neighboring villages will accept a stranger showing symptoms into their village.

Assumption 3: The village has a constant population for the duration of disease. No people are born or die from causes other than the disease during the course of the disease.

***Justification*:** A remote Indonesian village will not have very much interaction with the outside world in the first place, barring nearby villages. However, after the village is discovered as having a dangerous infectious disease, there will be extremely minimal interaction between the village and nearby villages or islands, as people will be highly discouraged from entering or leaving the village. It is highly unlikely that a person would enter a village afflicted by a dangerous disease. In addition, people inside the afflicted village will have difficulty leaving; as half the village is infected, they cannot help transport others to nearby villages/islands, not to mention that the authorities of other villages/islands will not allow people from the afflicted village to enter to prevent spread of the disease.

Since this is a potent disease because a near majority of the village is infected, and 15 people were killed in the last week, it is extremely more likely that the disease will kill a given person rather than other causes such as old age. In addition, it’s just as unlikely for new births to occur during this time, as previously mentioned, because many people are currently sick and displaying symptoms.

Assumption 4: Rates of transmission from Susceptible to Infected and Infected to Recovered are relatively constant.

***Justification*:** No information is given that would suggest that rates of transmission change or fluctuate over time. In addition, many other diseases have relatively constant rates of transmission, allowing people to create accurate mathematical models of them. Therefore, we will assume that this disease behaves in a similar manner.

Assumption 5: The demographic of the villages in question is representative but not necessarily proportional to the demographic of Indonesia as a whole.

***Justification*:** Due to lack of information, we cannot directly determine the demographics of the village, so utilizing the country’s whole demographic is the most accurate statistic we can use estimate this demographic. We do need to take into account the demographic of the village in our model, because it affects interactions between people as well as degrees of susceptibility.

Assumption 6: The area of the village mirrors the average population density of Indonesia as a whole.

***Justification*:** The area of the village is an unknown but necessary variable to consider in our model. Thus, basing the area off Indonesia as a whole will result in the most accurate and reasonable assumption of the value of this variable.

Assumption 7: Disease is spread through human interaction rather than a vector.

***Justification*:** Diseases that incorporate vector-based transmission tend to be slower than those that are transmitted through human interaction. The fact that nearly 50% of the population is infected indicates that this is a disease that spreads quickly -- certainly too quickly for it to be a vector-based disease. In addition, there is no information given whatsoever that justifies a vector-based transmission model.

4. Model

4.1 Model I—Standard Condition

4.1.1 Goals of Model

This model was designed to address the following concerns:

- Classify the type and severity of an epidemic that is taking place in a small village in Indonesia
- Determine if an epidemic is contained or not based on the set of limited information provided and what could be assumed
- Distribute various limited sources of aid to contain the disease most efficiently and effectively
- Simulate the effect of the disease under different circumstances through repeated run-throughs of the model under different conditions.

4.1.2 Summary of Model Program

We wrote a program that modelled the spread of the disease through the starting population of the village as described in the problem. Many factors, including transmissibility, avoidance, etc. were unknown but taken into consideration, and will be discussed further later in the paper. We created a village with 300 inhabitants, slightly less of whom were infected with the disease and the rest were healthy but susceptible. We calculated the rates of interaction between infected and healthy people every day to model the transmission of this disease. The program would continue to run until the number of total infected people reached zero. This program was run hundreds of times, under many different starting values of our unknowns, to observe overall trends as well as optimal responses. The full code, written in Java, can be found in the Appendix.

4.1.3 SIR Model Influences

The foundation of our models comes from the SIR model, used widely in epidemiology. The SIR model groups people into categories by their relationship with the infectious disease. S stands for susceptible, which is a person who is able to contract the disease. I stands for infected, which is the number of people who currently have the disease. A variation is the SEIR model, which includes Exposed (E), those who have the disease but are in a latent period and do not show any symptoms. For simplicity, we grouped E with I: after catching the disease, a given person would have an incubation period of a nonnegative number of days, after which they would start displaying symptoms; however, we considered both of these people as “infected”. Infected people can go in one of two directions. If they recover from the disease, they go into group R, which stands for Recovered. These people gain immunity to the disease and will survive after the infection leaves their body. The rest of the infected people will eventually die as a direct result of the disease.

The SIR model is useful because it allows us to calculate rates of change for each of these categories. From this, we can determine how the infection is moving by seeing the movement of people between categories. Define S, I, and R as the number of people in each of these categories, and N the total number of people. Also, s, i, and r are the ratios of people in each of these categories; $s = S/N\ i = I/N$, and $r = R/N$. We also define $\beta = r \cdot c$, where r is the transmissibility, or the probability that an infected person will infect a susceptible person in a given interaction, and c is the average rate of contact between susceptible and infected individuals. Then, we have $ds/dt == -\beta si$, $di/dt = \beta si - \nu i$, and $dr/dt = \nu i$, as illustrated in Figure 1. We know that an uncontained epidemic occurs when the number of infectious cases is increasing, so we must have $\beta si - \nu i > 0$ for this to be the case.

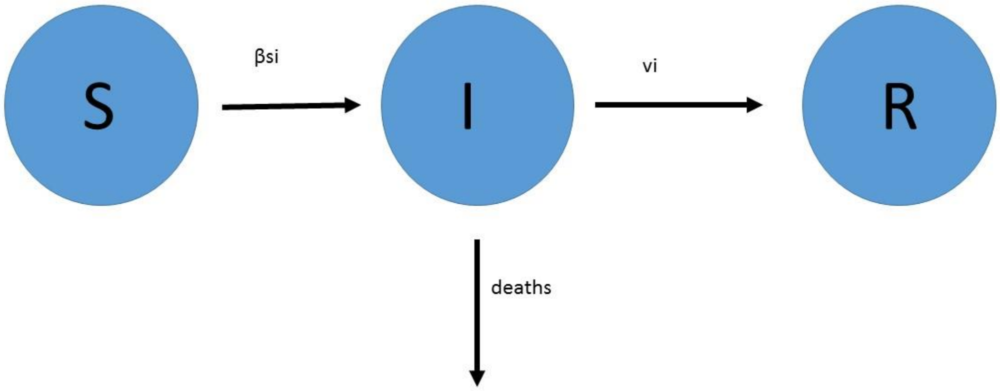


Figure 1. Pictorial representation of the SIR model.

In our actual model, we split people into these different categories: susceptible, infected, immune, and dead. For immunity, we also considered passively immune people, those who were naturally immune to the disease without having come into contact with it, and we set the percentage of those people to be 0.1%. If an individual recovers from being infected, he is also immune. We allowed people to interact with each other every day, spreading the infection, and repeated until the number of infected people became 0, either because they all died from the disease or the disease itself killed itself before that.

4.1.4 Spread of a Rumor Model Influences

The Spread of a Rumor model aims to mathematically predict the spread of a rumor given a particular population of people. In this population, there are only spreaders and ignorants -- those who know the rumor and those who do not. This model utilizes the assumption that this rumor will spread only through the interaction between a particular spreader and an ignorant. Given a population of N people, K of which are spreaders, this model predicts that the expected number of new people that hear of the rumor in one day is calculated as: $K - \frac{K^2}{N}$.

We observed a striking parallel between the spread of a rumor and the spread of a particular disease, so we thought we could employ this equation in our model, albeit with a few modifications. The spreading of the disease in Indonesia is very much like the spread of a rumor in a population. They both need to be spread by human contact, and rather than spreaders and ignorants, we have infectors and susceptibles. In our modified equation, we have K infectors and a total of N people.

Right now, the equation $K - \frac{K^2}{N}$ would represent the total number of new infected people per day. First of all, we must adjust for the fact that an interaction between an infected and susceptible does not guarantee the disease being spread. In the rumor model, each human interaction results in an ignorant becoming a spreader. In this way, the rumor can be considered to have a 100% success rate. Thus, we need to account for the fact that not all human interactions between an infected and susceptible person will cause disease. To do so, we multiply our formula by the transmissibility value of our disease, represented by i and expressed as a decimal between 0 and 1. This value is the probability that a human interaction between an infected and susceptible person results in the susceptible person becoming infected.

In addition, there are other factors we would like to consider to determine our modified formula. Area is such a factor. In an infinitely large space, the probability of interacting with another human is infinitely small. A small area will result in many human interactions, as it is much more likely for two people to cross paths in a small area. However, we are not given the area of the island. Thus, we utilized the population density of Indonesia to come up with our estimation. The population density of Indonesia is 122 people per square kilometer. Thus, a village of approximately 300 people will have an area of approximately 2.5 square kilometers. To account for area, we take the baseline area as determined from the population density, 2.5 square km, and divide it by the actual area. We utilize a quotient based on the previous logic regarding the relationship between space and human interaction. Large areas will result in this quotient being less than 1, decreasing the expected number of infected people. Small areas will result in this quotient being greater than 1, increasing the expected number of infected people.

The expression we have so far is $\frac{2.5}{Area} * i * (K - \frac{K^2}{N})$. We also have to take into account the avoidance rate, that a given susceptible individual will try to avoid an infected individual showing symptoms. We set this variable as s, and while testing our model, we increased and decreased it to see what effect avoidance rate would have on transmission. An avoidance rate of 0, or 100% avoidance, would result in no new infections.

Our formula as of now only determines expected number of new infections per day. We divide this by the total number of susceptibles , S, to find the probability a given susceptible is infected on that day. Set this expression as X for now.

$$X = \frac{\frac{2.5}{Area} * i * (K - \frac{K^2}{N})}{S}$$

Now we want to find the probability a given infected will infect a given susceptible. The probability is not simply X/K, where K, is the number of infected. Imagine if the probability X that a susceptible will be infected at the end of the day is ½, and there were 2 infecteds. Would the probability that each infected would infect the susceptible be ¼? No, because the two cases are independent of each other. The chance the susceptible is infected at the end of the day in that case would be $1 - (\frac{3}{4})^2$, which is the probability that at least one infected infects the susceptible. This is clearly different from our original assumption that the probability the susceptible will be infected is ½. Using similar logic, we can derive an accurate formula for the probability P that a certain infected infects a certain susceptible.

$$1 - X = (1 - P)^K$$

$$P = 1 - (1 - X)^{\frac{1}{K}}$$

This probability P is defined as the probability that a certain infected person infects a certain susceptible on one day. We can incorporate this into our model, finding the probability of each interaction causing a disease each day, to study the growth of infected people. We used a formula to study the spread of rumors and modified it to work with disease spread as well.

4.1.5 Analysis of Variables Used

Transmission rate: This refers to how easily a disease is able to spread from someone who is already infected to someone who is susceptible to the disease. Mathematically, it is defined as the probability that any interaction between a healthy, non-immune person and a victim of the disease will result in the healthy person contracting the disease. As such, this value ranges from 0 to 1, with a lower value meaning that a disease only is transmitted on rare occasions, and a higher value meaning that the disease is passed on at nearly every interaction between an infected and susceptible person.

Latent Period (days): When someone first contacts a disease, there is a time period during which they are infectious, that is they can transmit the disease to others, but are not displaying symptoms. While someone is in this latent stage, they are extremely dangerous because they might continue to engage in regular interactions with others while both they and those that they come in contact with are oblivious to the fact that they are exposing others or are being exposed to a deadly disease. For our purposes, the latent period is measured in days, with an average latent period falling roughly between 1 and 3 days.

Infected Period (days): After someone contracts the disease and completes the latent phase, their disease moves into the next stage, the infected period. While in this state, a person is still capable of transmitting the disease to others, but is also displaying symptoms. As a result, they are still dangerous to others, but unlike during the latent phase, they are obviously sick and can be avoided by healthy people. This means that the probability that they will infect any particular susceptible person should be reduced, since the number of interactions that they would have would also be lower. A typical range for the infected period would be roughly between 4-10 days, but since we are not provided with an expected value, we had to use multiple values and a series of simulation runs to gather some comprehensive results that encompassed this whole range.

Removal Rate: After someone completes both the latent and infected periods, they will pass out of the cycle of the disease, and will either survive or be killed by the disease. The removal rate refers to the probability that someone survives and gains immunity to the disease instead of dying. Interestingly, this value is known to be the reciprocal of the duration of the infected period showing symptoms. This implies that the longer amount of time one spends sick, the less chance remains that they will eventually survive the infection. As such, a typical removal rate would range from .05 to .25.

Avoidance Constant: Amongst a population that is well aware of the effects and concerns of infectious disease, the collective response to outwardly sick people would be to avoid them. However, in a more remote area of the world, such as Indonesia, there is the possibility that interactions with the sick would not necessarily curtailed to the same degree as otherwise. This means that we need to take into account the probability that an interaction that would occur under normal circumstances would not occur if one of the participants was exhibiting symptoms of disease. In locations where knowledge of basic epidemiology is minimal, it is possible that the value of this constant would be very close to 1, meaning that nearly one hundred percent of interactions would occur just as they normally would. Since this value refers to a probability, it falls on a range of 0 to 1.

Avoidance Rate: This value is equal to one minus the avoidance constant, and is the probability that any particular interaction between an outwardly sick person and a susceptible person would not occur due to them avoiding each other.

4.1.6 Model Building

The model we developed was based upon some of the key concepts from the SIR and a few other mathematical models that we found throughout the research process. The overall concept behind our simulation was to model the situation using as many individual processes as possible. We simulated the simplest interactions between the members of the village using calculated probabilities and random chance and then viewed how this affected the community as a whole. In doing so we broke down the epidemic into a two levels, the village and the individual. On the scale of the entire village, we conducted the processes that would affect everyone, including the calculation of the overall general percentage of interaction possibility, namely the probability that an infected person and a susceptible person would come in contact and would result in a successful disease transmission on any given day. This was calculated using the equation

$$P = 1 - (1 - (\frac{\frac{2.5}{Area} * i * \left(K - \frac{K^2}{N}\right)}{S}))^{\frac{1}{K}}$$

as derived earlier. After this value was calculated, it would be applied for every single infected person in the population, each of which would then in turn apply it to every single susceptible person in the population. A random number would be generated for every one of these interactions based on this probability, and if it indicated a successful transmission of the disease, the state variables associated with the susceptible person would be modified to reflect his new status as infected. Upon becoming infected, there were another series of processes which would apply to a Person object every time the Village looped through each day. For example, a Person object would have to keep track of how many days had passed since its initial infection, and how that compared to its own randomly generated values for latent period and infected period. Then, upon passing from latent to infected, the object would be modified to reflect its symptom status, which would affect the probability that the person would transmit the disease to someone else. After the infected period is also completed, the Person object would again apply a probability, specifically the removal rate. This event would decide whether a person would die or become immune after having the disease. Moving back to the level of the village, this overall process would occur once per day, and would happen again and again until the number of infected persons in the population reached a value of 0. After each day, the model would output the statistics observable during real-life epidemics, such as number of people infected, dead, healthy, immune, and susceptible as a function of days to a series of Excel spreadsheets. Then, by analyzing the spreadsheets using more computer driven algorithms, and through our own manual methods, we were able to determine many important characteristics regarding the disease in question. In this way, our model provided us with simulated values for an epidemic under any set of circumstances as a function of days, which we could then modify hundreds of times to determine how modifying certain factors would affect the village as a whole.

When designing this model we decided that the most effective way to approach the situation of an epidemic where we had extremely limited knowledge of any specific details was to base as many parts of the program on probability. However this meant that the model would most likely return different values for the same set of input data, which we compensated for by running the simulation hundreds of times. By doing so, we were given thousands of data points to work with, which we could not analyze entirely by ourselves, which prompted us to write another program to analyze any run of the simulation and mathematically calculate a severity value for the epidemic. This value was based on a formula that we devised ourselves:

$$||\frac{10m^2Di}{n^3\Delta x}||$$

Where:

m = max # of infections at one time

D = total deaths at end

i = # of day when rate of infection is min (i.e. steep drop off in infected)

n = total population at beginning

Δx = total # of days

the double bars denote rounding to the nearest integer.

This formula returns a number between 0 and 10, where 10 is the most severe possible epidemic and 0 is the least severe. In this equation, the value for the maximum number of infected people, number of people dead after the disease is no longer being monitored, and the time in days at which the rate of new infections reached a minimum value are offset by the total number of people in the population and the duration of the epidemic. We made sure to balance the effect of each variable such that the more important constants, specifically the maximum number of infected persons had a more extensive role in determining the severity value. This was accomplished by squaring the value in the numerator, and then likewise cubing the population size in the denominator. By using this formula to evaluate the severity of our simulated epidemics, we were able to determine what set of initial parameters resulted in the most effective handling of the situation. Then, we were able to decide what allocation of resources would result in the parameters falling into a similar pattern, which would in turn cause the epidemic to resolve itself as quickly as possible.

Overall, the extensive use of probability in our model is the major factor for why it able to provide us with such accurate simulated epidemics. Through research and trial and error, we were able to come up with mathematical representations of the probabilities of events occurring based on the same 5 original parameters. Then, using the immense power of computer programming, we were able to repeat these simulations hundreds of times, and then analyze the data to determine the most effective solutions. In this way, we harnessed the power of probability to model the complex processes that are disease spread and human nature.

4.1.7 Model Results

By analyzing the output files from our model, we have been able to come to a few conclusions regarding the key questions that we set out to answer.

Using the number of villagers affected by the disease and the time over which the disease was active, we were able to determine the type and subtype of the outbreak. There are two major types of outbreaks; common source outbreaks and propagated outbreaks. Common source outbreaks occur from exposure to a single source, while propagated outbreaks are transmitted from person to person. They consist of two subtypes; point source outbreaks, which begin and end very quickly, and continuous source outbreaks, which occur over long periods of time. In addition, intermittent outbreaks can be common source outbreaks, but they usually arise from many different sources. Based on the data we gathered from running our model many times over, we generated a graph showing a series of the generated epidemic curves. As can be seen in Figure 3, they all rise sharply until they reach a maximum value and then fall at a slightly lower pace until the number of cases reaches almost zero. These characteristics imply that this disease is following the course of a point-source outbreak, since they follow many of the general traits that one would expect when viewing the epidemic curve of a point-source epidemic curve.

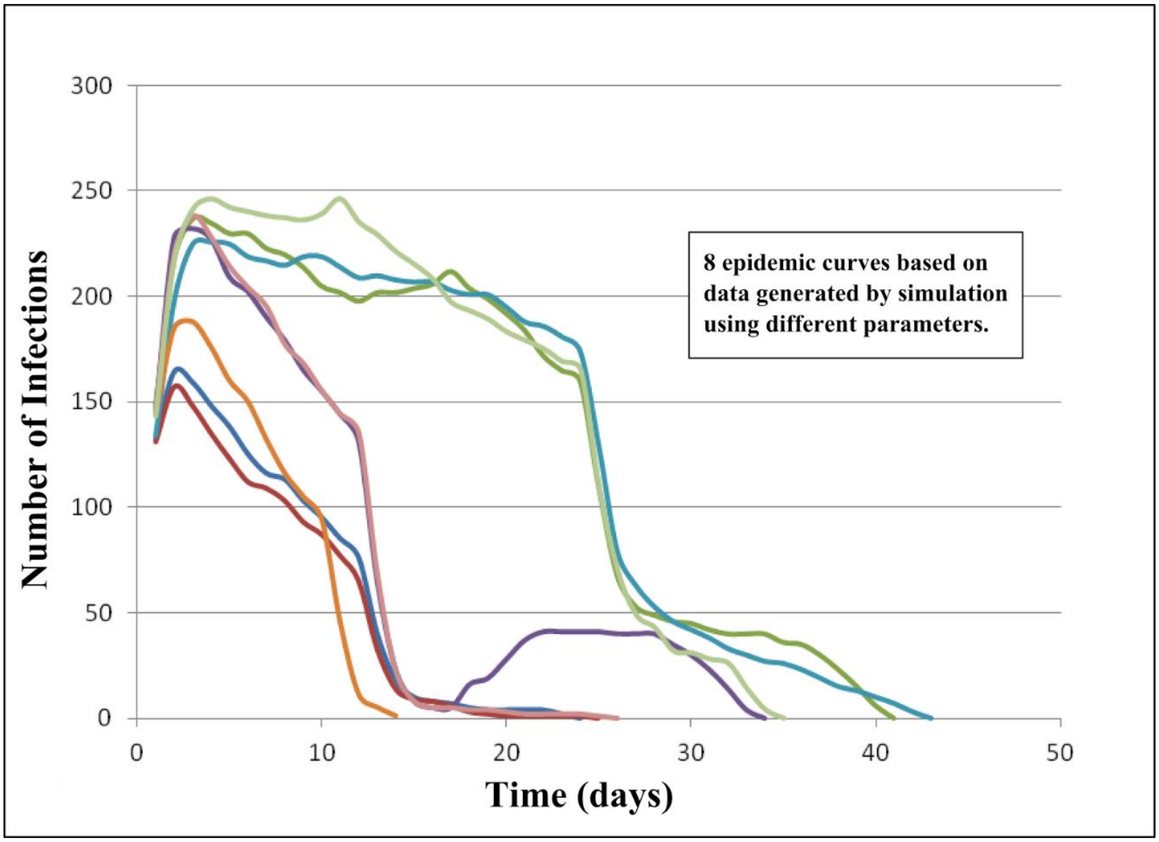


Figure 3. Graph of total number of infections over time for 8 different epidemic curves. This shows that regardless of initial parameters, it fits a point-source epidemic.

Using our derived formula from earlier, we determined the impact that the changes in variables had on the severity of the infection. The severity depends highly on the number of maximum infected at one time, the total number of dead, and the length of time it takes before a sharp fall in infected occurs. We analyzed a sample of 15 data points each with their own custom parameters, as shown in Table 1. We can use this to observe any common trends in severity. From this table, we see that the overwhelming majority of tests have a severity between 1 and 4. However, the duration of the infected period when an individual shows symptoms seems to be the most important factor in determining severity, with severity increasing as the duration of the period increases. This makes sense, because as duration of infection increases, its reciprocal, the recovery value, decreases. This makes it more likely that more individuals will die. Also, the drop-off in infected will occur later in the simulation because the majority are infected from the beginning, so there is a direct correlation between duration of infection and drop-off time.

Finally, the max number of infected will be greatest with a larger duration because the longer time means more will be infected at one time. After running these trials, we conclude that most of the time, the severity of the disease is not very high; it usually runs its course rather quickly and not too many people die. However, the changing of a key value, duration of infection, which we are unable to determine outright, can change the severity of the infection.

Trial ID	Calculated Severity
1111	1
1112	1
1113	0
1117	3
1118	3
1119	3
1121	3
1122	4
1123	0
1123	0
1124	3
1124	3
1126	2
1127	2
1132	8
1134	8
1137	2
1138	2
1140	1

1145	4
AVG.	2.65

Table 1. Calculated severities of 15 different trials of the infection. The parameters set for each trial can be found in the appendix.

Another question we were looking to answer was whether or not this epidemic was contained. If the rate of new infections is negative, then the epidemic can be said to be self-containing, meaning that if allowed to run its course, it would eventually leave the population of its own accord. As is seen in the epidemic curves shown above, this is the case for all of the output generated by the simulation, which included runs that were based on unique sets of input values. Since we saw this trend in every single curve that we analyzed, we can conclude that the epidemic that has been observed in Indonesia is contained.

The final goal of our model was to determine what would be the most effective methods to deal with the epidemic given our limited resources. Our options range from transporting victims and treating them with external aid to letting the disease continue acting by itself. Based on our conclusions that the disease is point-source, not very severe, and self-contained, we have determined a plan that logically makes sense. In the beginning of the outbreak, doctors should be brought in to treat the initial cases and prevent susceptible people from getting infected. They will also inform the susceptibles on ways to avoid infection. By doing so, we are able to prevent the disease from spreading too much. Although the doctors will do these two tasks, it is unlikely that they will be able to stop the disease so early. This is because about 150 citizens are already infected. Thus, there will be an inevitable increase in number of infected people, due to the sheer size of the infected population. During this time, doctors should continue focusing on preventing the infection of susceptible citizens. Although there is an increase in the number of infected people, this value will peak, at which point doctors should switch their focus to only curing patients. The disease is already heading downwards, and with the aid of the doctors, it can be ended quicker. This is the best plan, as it is the most effective and efficient, in terms of time and money. For example, a containment center for specific infected individuals is unnecessary as it is so expensive and the outbreak is so short-lived. Drugs, also will not be developed fast enough and would be too expensive during the course of the disease.

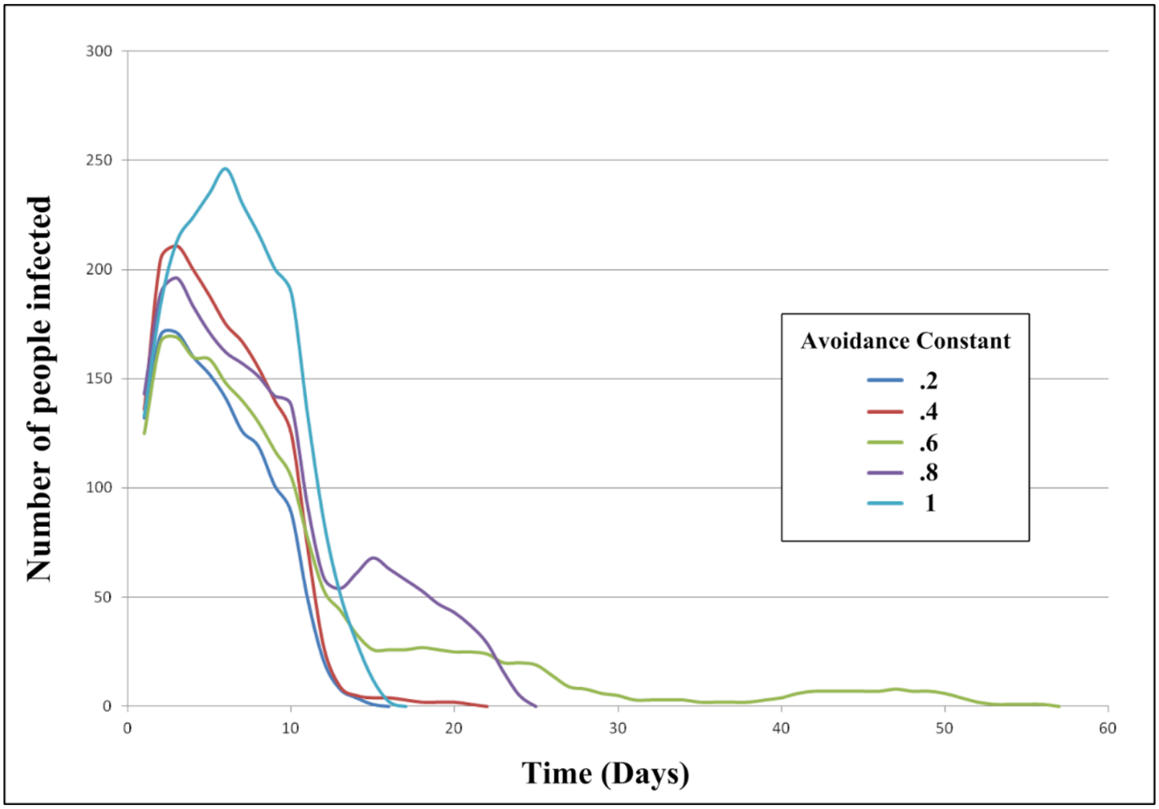
Implications of Model

4.2.1 Recommendations for Center of Disease Control

The Center of Disease Control, although they want to help anyone sick if possible, has the ultimate goal of preventing the spread of disease, which if not prevented could cause dangerous epidemics and pandemics. As they say in their mission statement, their primary goal is to “fight disease and support communities and citizens to do the same”. Therefore, the reasoning behind our recommendations is based on preventing as many people from being infected by the disease as possible.

To begin with, we strongly recommend that the first order of business is to educate the inhabitants in the village to stay away no matter what from a sick individual, in order to prevent the spread of disease between infected and susceptible people. This will increase the avoidance rate in our model, which determines how well a susceptible individual will avoid an infected individual showing symptoms. Our model demonstrated that avoidance had an extreme impact on severity factor. This makes sense, because the disease spreads through human interaction with an infected, so less human interaction for an individual means less chance of becoming infected. Since the disease is occurring in a rural Indonesian village, it is probable that the native inhabitants do not have a modern view on disease control. Perhaps they, like some other indigenous people, believe the disease is caused by the anger of the gods or other supernatural factors. At the least, their knowledge of the science of disease transmission is nowhere as much as what we experts have. The best way that we can increase the chance that susceptible people avoid infected people is to educate them on the basic nature of infection -- that disease is transmitted from infected people to healthy people by human contact. Since they do not want to get sick themselves, they will avoid infected people to remain healthy themselves. We need to absolutely emphasize the point of avoidance as soon as possible in order to prevent the further growth and spread of this disease. In our models, if the avoidance constant (which is defined as (1 – avoidance rate)) got closer to zero (i.e. the susceptible avoided the infected showing symptoms almost 100% of the time), the number of total infected people fell drastically and the disease died out much sooner. As you can see in Figure 2, the lowest avoidance constant, .2 (meaning that susceptible avoided infected 80% of the time), caused the lowest total number of people infected at one time. In addition, the disease died out much sooner versus the other run-throughs that set a higher avoidance constant.

Figure 2. Graph of number of people infected over time in a village with variable



avoidance constants.

The second order of business is to ensure that the disease does not leave the village, taking whatever preventative measure necessary. Our models strongly suggest that after a certain number of days, the disease will run its course, killing itself and leaving no infected people left. However, it still is a very potent disease, with our models suggesting a final mortality rate of usually ½ or larger. Because of this, if the disease accidentally spread into nearby islands or other countries, it could cause a global pandemic with a huge death rate. Therefore, even if it means we cannot send doctors and other aid workers into the village, we must ensure that no contact whatsoever occurs between the village and the outside world that would cause the disease to be spread outside the village. It may harm the inhabitants of the village more in the short term, but this damage pales in comparison to what may happen if the disease is unleashed to the rest of the world. A complete quarantine of the village would be the best course of action. It would be expensive, but it would potentially save many human lives.

As spending money on the quarantine is imperative, some other potential resources will not be necessary. In addition, we will not need research on novel drugs or serums to cure the disease. Our models always show the disease running its course in a short duration of time, usually less than 50 days, if contained in only the village. This is much less than the average of 12 years to completely develop a new drug.

A vaccine will also not be necessary for this disease. At the end of our model, after the disease has killed itself, all the remaining people are either dead, or immune to the disease. Thus, they are naturally protected against the disease already.

A containment center will also not be necessary for this disease. Although we can feasibly contain the entire village from the outside world, it isn't feasible to detain and quarantine all of the currently infected people, nearly 150 people. As long as we preach avoidance, and prevent the spread of the disease outside the village, such measures be superfluous.

4.2.2 Further Questions

A multi-national research team has just returned after gathering information about the disease in the village for 7 days. We can only ask them three questions. We believe that to best improve our model, we should ask questions for constant values that we are uncertain of and that right now we have as variables. In addition, they should be values crucial to the experiment and can be used in several different equations (for example, the recovery rate is the inverse of the duration of infection, so asking about that gives us two unknown pieces of information). However, we should ask questions that are able to be easily observed (i.e. we shouldn't ask them questions about transmissibility rate, etc. that would require experiments to determine). After careful consideration, we have determined these three questions to ask along with justification:

1. *How many days after showing symptoms does a typical victim die or recover?*

As previously mentioned, this question allows us to find the typical duration of infection, which is a crucial value that thus far we have been estimating. It is a value vital to our model, as different durations of infection will cause the course of the disease spread in the village to be longer or shorter. In addition, since the inverse of duration of infection is the recovery probability v, we will know from this the potency of the disease. Since (1-v) is the probability of death, if the duration of infection is longer it also affects how many people recover versus how many die. It is also a value that can easily be observed through solely observation rather than needing experimentation. A problem we may run into is if the duration of infection is longer than the 7 days the researchers are staying. In this case, we will not gain any concrete information from this question, but it will still allow us to rule out durations of infection lasting less than 7 days.

2. *What is the area of the village in square kilometers?*

As previously mentioned, a key assumption was that the area of the village, and thus the population density, is proportional to the population density of Indonesia as a whole. However, we have no way of determining this fact instead of assuming it. This is also a rather vital data point, because our formula for probability of interaction between infected and susceptible is based on the size of the island. If the island is much larger/smaller than our assumption, our entire model will be inaccurate. This is also a question that is easy to observe.

3. *What are the symptoms of the disease?*

We are not able to directly find the transmissibility probability, barring direct testing of people infected with the disease, which we are obviously not allowed to do. However, knowing the symptoms of the disease will give us crucial insight in indirectly estimating this probability. If the symptoms include sneezing, coughing, diarrhea, or others that would help transmission between people, the transmissibility would be higher. In addition, the severity of the symptoms would determine the rate of avoidance, both because the sick would more likely to stay at home, reducing interactions, and because susceptible will be more likely to avoid someone extremely and obviously sick. Again, it is also a piece of information findable by direct observation.

4.3 Model II—Revised Given More Information

The findings of the multi-national research team concluded the following of the disease:

- It appears to spread through contact with bodily fluids of an infected person
- The elderly and children are more likely to die if infected
- A nearby island is starting to show similar signs of infection
- One of the researchers that returned to your capital appears infected. We assumed that the capital city in this case was Geneva, Switzerland, as that is the headquarters of the IWHO.

Based on these facts, we will need to update our model, not only for the village, with the additional information about the disease behavior, but also for the new environments of the island and the capital. In addition, we can take into account the responses to our questions asked by assuming constant, arbitrary values for those previously unknown values. For instance, we will assume that after learning about observations, we find that:

- The average duration of infection with symptoms is 10 days.
- The area of the village is 2.5 square kilometers.
- Symptoms were not able to be determined; however, like we learned above, it spreads through contact with bodily fluids, which will help us with determining transmissibility.

4.3.1 Changes to Model

Once provided with additional situational information, we were able to update some aspects of our model to reflect a more accurate view of the situation in the Indonesian village. Changes ranged from specifying the effect of age on the progression of the disease, the specific method of transmission, and the inclusion and testing of some extra scenarios where the behavior of the disease would be relevant.

One such change involved gaining some additional insight into how the disease is transmitted from one person to the next. According to the researchers, this mysterious disease spreads through contact with bodily fluids. Since Ebola spreads in a very similar manner to the disease we are examining, we used the transmissibility of Ebola as the default transmission risk in our code. The transmission risk P is calculated as $R_0/(C*D)$, where R_0 represents the basic reproductive number (average number of infections caused by an infected person), C represents the number of people the infected individual has come in contact with, and D represents the duration of the infection. We tried to determine the value of P for Ebola from existing scientific literature; however, we could not find the desired value. In order to determine this value, we set up the proportion shown below, obtained by solving the above equation for C and assuming that D = 1 year. HIV was chosen as the "comparison" virus because it had readily available data for both R_0 and P.

$$R0/P\ of\ Ebola\ =\ R0/P\ of\ HIV$$

The R_0 values of the two viruses and the P value of HIV were identified and inputted to produce the following proportion, where P is a percentage:

$$4/11.5\ =\ 1.87667/P$$

Solving this proportion gives a value of P = 5.3954% for Ebola virus.

The second change regards how people of different ages are affected by the disease. We were told that the elderly and children were more likely to be killed by the infection, which we then proceeded to account for in the model by decreasing the removal rate for the those aged 14 and under and 55 and over by 25%. This should make it so that when children and the elderly progress to the end of the infected period, they should have slightly reduced chance of surviving the ordeal when compared to individuals in their prime.

The final change to the model involved being able to modify the number of total and infected persons to start off with in the population. This was necessary because now we needed to consider the possibility that there were people infected with the disease in other uncontaminated populations. To model these scenarios, we needed to change the initial parameters such that they could reflect the environment of another nearby island, and then our home country of Switzerland. While we already had access to the majority of the variables, we had to change the initial values of the infected and total populations.

4.3.2 Updated Model Results

The results from version 2 of our model have shown us more information about the mysterious disease that we did not know previously. We were able to determine the effect of special demographics, transmissibility of the infection, and circumstances of the epidemic on the spread of the disease.

First off, the addition of the special removal rates based on age seemed to have a limited effect on how the disease behaved in the village. We noticed that apart from the number of infections changing at a lower rate and leveling out sooner, there was almost no difference between how the disease reacted to the situation before we included age and how it did after. This implies that it was not a huge factor on how the disease would spread.

Second of all, the new knowledge that the disease spreads through bodily fluids did alter our transmissibility rate. It lowers the transmissibility rate, as it becomes harder to transmit the disease if it only spreads through bodily fluids. Thus, it causes the disease to fall out faster. Those who are infected cannot spread it to someone else fast enough before they die.

Finally, we used the model to come up with a prediction for how the disease would behave in two new scenarios, an uncontaminated island and Switzerland. In both cases, we assumed that there were a minimal number of infected cases, specifically 1-5 people. However, when the model was run, we found that since the initial size of the infection was so small that every single victim reached the end of the infection period before infecting another person. This meant that in both cases, the island and Switzerland, the epidemic contained itself almost immediately. Because of this, we tried increasing the number of people who were infected at the beginning of the simulation. In spite of this, we found that the majority of infected persons on the island either died or became immune before spreading the disease, meaning that the epidemic was still contained. In

Switzerland, we reduced the transmissibility of the disease due to the temperature change, and encountered similar results when we ran the simulation with 1 index case, and with almost half the population infected. Therefore we can conclude that the disease in question does not pose a threat to human society when not heavily concentrated.

4.4 Strengths and Weaknesses of Model

Strengths

1. Our model makes relatively few assumptions that weren't directly stated in the problem. Many of our values, such as transmissibility, duration of infection, avoidance rate, etc. were left as variables which we could modify as we liked. Thus, we could test the spread of the disease under a countless number of preexisting conditions, which works well given that we know very little about the actual behavior of the disease.
2. Our model also takes into account the random chance of real life. In real life, interactions between people do not happen nearly as ideally as many models predict. There are a lot of random factors involved, so we designed our model incorporating pseudo-random chance. For example, when we analyzed whether a given infected would infect a given susceptible, we calculated the probability whether this would happen, and we used pseudo-random chance to determine whether it would actually happen in each relationship between any infected and susceptible.
3. Our model's power comes from its flexibility. Given more concrete information about the behavior of the disease, we can easily plug those values into our current model and thus create an accurate model of those specific conditions.

Weaknesses

1. Basic values of rates of spread, such as transmissibility, infection probability, duration of infection, and recovery rate must be assumed instead of having concrete numbers to use.
2. Our model is adapted specifically to this particular situation in this village, and cannot be easily applied to other possible epidemic cases.
3. We made multiple assumptions about the information of the village, such as its population density, area, and demographic. If these assumptions were not valid, our model would not be completely accurate.
4. Since the majority of our model is based on probability, any one run cannot be expected to be near 100% accurate. However, the cumulative results of a large number of runs can be used to make broad generalizations about the disease.

5. Conclusion

The results from our model indicate that the duration of the infective period and avoidance are the most important parameters in controlling the spread of the disease. If the infective period is relatively short, then it becomes much easier to eradicate the disease, as infected people will be less likely to spread it to others. In addition, increased avoidance leads to reductions in both the number of days the disease takes to be eliminated and the maximum number of infected people. As shown in Figure 2, this is a rather short-lived disease, as it disappeared within a month in most scenarios, and in all cases, it disappeared less than 2 months after it began.

The effects of the additional criteria modified the results of our model in a noticeable way. Assuming approximately a .05 value for transmissibility, based on that of Ebola, the disease dies out only more quickly, as it becomes even harder to spread from an infected person to a susceptible. The additional assumption of age having a role on death also didn’t matter very much, as age didn’t play a large role in our model. The spread of the disease in the nearby island can be halted through isolation just as easily as it was in the village. Regarding the researchers bringing back the disease to their native country, that had a very minimal effect as well, as the model showed the disease dying out nearly immediately given a small original infected size.

Despite the obvious potency of the disease, the fact that it is only impacting a small village means that it can be contained relatively easily. Even when it is considered that the disease has spread to a neighboring island and has been contracted by a humanitarian worker, it is easy to prevent the disease from spreading further and becoming a global pandemic.

6. Synopsis of Findings to News Outlet

What started as a routine humanitarian mission to a small Indonesian village has been receiving worldwide attention, as the citizens of the world begin to hear about this deadly disease outbreak affecting nearly 50% of this village's population. With 15 villagers already dead, people are beginning to question if this disease is going to spread to their own homes. With such limited resources, the world can only wait to see what this small Indonesian village will do. A report from the team is shown below:

Right now, the most important mode of defense is education and knowledge. We will continually have updates about the behavior of the disease, and appropriate recommendations and precautions to take. As of now, the disease that has afflicted this small Indonesian village seems to transmit itself solely through human contact. We recommend that everyone stay inside and avoid contact with strangers. At early stages of infection, it is not a guarantee that one will show symptoms. Thus, one can never be sure if a stranger is infected or not.

Given the isolation of this small Indonesian village, a simulation of the disease spread has indicated that it is unlikely the disease will spread to those outside the village. To increase the likelihood that it does not spread outside the village, we have already begun restricting movement in and out of the village, putting the village on quarantine to ensure that the disease burns itself out. We cannot emphasize how important it is to avoid contact with infected people. The disease cannot spread if there is no one to spread it to. If human interaction is greatly reduced for the time being, we estimate that this outbreak can end in 15-25 days.

The team and Indonesian government are still working hand in hand, hoping their efforts will be enough to save the village, and if necessary, the world.

7. References

Althaus, C. (2014). Estimating the Reproduction Number of Ebola Virus (EBOV) During the 2014 Outbreak in West Africa. PLOS Currents Outbreaks, 1, 1-9. Retrieved from http://currents.plos.org/outbreaks/article/estimating-the-reproduction-number-of-zaire-ebolavirus-ebov-during-the-2014-outbreak-in-west-africa/ on November 2, 2014. doi: 10.1371/currents.outbreaks.91afb5e0f279e7f29e7056095255b288.

Basic Reproductive Rate (n.d.). Retrieved from http://practice.sph.umich.edu/micphp/epicentral/basic_reproduc_rate.php

Chemaitelly, H., Awad, S., & Abu-Raddad, L. (2013). The risk of HIV transmission within HIV-1 sero-discordant couples appears to vary across sub-Saharan Africa. Epidemics, 6, 1-9. doi:10.1016/j.epidem.2013.11.001

Ebola: Mapping the outbreak. (2014, October 30). Retrieved November 2, 2014, from http://www.bbc.com/news/world-africa-28755033

Epidemic curves. (n.d.). Retrieved November 2, 2014, from http://www.med.uottawa.ca/sim/data/Public_Health_Epidemic_Curves_e.htm

Epidemics, Pandemics, and Outbreaks of Contagious Diseases. (n.d.). Retrieved November 2, 2014, from http://www.webmd.com/cold-and-flu/what-are-epidemics-pandemics-outbreaks

Fedewa, N., Krause, A., & Sisson, A. (2013). Spread of a Rumor. Retrieved from Society for Industrial and Applied Mathematics website: http://www.siam.org/students/siuro/vol6/S01182.pdf

Jones, J. H. (2007). Notes on R0. Retrieved from Stanford University website: http://web.stanford.edu/~jhj1/teachingdocs/Jones-on-R0.pdf.

Mission, Role and Pledge. (2014, April 14). Retrieved November 2, 2014, from http://www.cdc.gov/about/organization/mission.htm

New Drug Development Process. (n.d.). Retrieved from California Biomedical Research Association website: http://ca-biomed.org/pdf/media-kit/fact-sheets/cbradrugdevelop.pdf

No, Seriously, How Contagious Is Ebola? (2014, October 2). Retrieved November 2, 2014, from http://www.npr.org/blogs/health/2014/10/02/352983774/no-seriously-how-contagious-is-ebola

Waterman, T. (n.d.). Brief General History of Ebola. Retrieved November 2, 2014, from https://web.stanford.edu/group/virus/filo/history.html

Where did HIV come from? (n.d.). Retrieved November 2, 2014, from http://www.theaidsinstitute.org/node/259

8. Appendix

8.1 Master File w/ All Test Runs

FILENAME	TRANSMISSABILITY	IMMUNITY RATE	LATENT PERIOD	INFECTED PERIOD	REMOVAL RATE	AVOIDANCE
1111.csv	0.4	0.001	2	12	0.083333	0.666
1112.csv	0.4	0.001	2	12	0.083333	0.666
1113.csv	0.4	0.001	2	12	0.083333	0.666
1114.csv	0.4	0.001	2	12	0.083333	0.666
1115.csv	0.4	0.001	2	12	0.083333	0.666
1116.csv	0.8	0.001	2	12	0.083333	0.666
1117.csv	0.8	0.001	2	12	0.083333	0.666
1118.csv	0.8	0.001	2	12	0.083333	0.666
1119.csv	0.8	0.001	2	12	0.083333	0.666
1120.csv	0.8	0.001	2	12	0.083333	0.666
1121.csv	0.8	0.001	2	24	0.041667	0.666
1122.csv	0.8	0.001	2	24	0.041667	0.666
1123.csv	0.8	0.001	2	24	0.041667	0.666
1124.csv	0.8	0.001	2	24	0.041667	0.666
1125.csv	0.8	0.001	2	24	0.041667	0.666
1126.csv	0.8	0.001	2	12	0.083333	0.333
1127.csv	0.8	0.001	2	12	0.083333	0.333
1128.csv	0.8	0.001	2	12	0.083333	0.333
1129.csv	0.8	0.001	2	12	0.083333	0.333
1130.csv	0.8	0.001	2	12	0.083333	0.333
1131.csv	0.9	0.001	2	48	0.020833	0.9
1132.csv	0.9	0.001	2	48	0.020833	0.9

1133.csv	0.9	0.001	2	48	0.020833	0.9
1134.csv	0.9	0.001	2	48	0.020833	0.9
1135.csv	0.9	0.001	2	48	0.020833	0.9
1136.csv	0.6	0.001	2	10	0.1	0.2
1137.csv	0.6	0.001	2	10	0.1	0.2
1138.csv	0.6	0.001	2	10	0.1	0.4
1139.csv	0.6	0.001	2	10	0.1	0.4
1140.csv	0.6	0.001	2	10	0.1	0.6
1141.csv	0.6	0.001	2	10	0.1	0.6
1142.csv	0.6	0.001	2	10	0.1	0.8
1143.csv	0.6	0.001	2	10	0.1	0.8
1144.csv	0.6	0.001	2	10	0.1	1
1145.csv	0.6	0.001	2	10	0.1	1
1146.csv	0.6	0.001	2	10	0.1	0.8
1147.csv	0.6	0.001	2	10	0.1	0.8
1148.csv	0.6	0.001	2	10	0.1	0.8
1149.csv	0.6	0.001	2	10	0.1	0.8
1150.csv	0.6	0.001	2	10	0.1	0.8
1151.csv	0.6	0.001	2	10	0.1	0.8
1152.csv	0.6	0.001	2	10	0.1	0.8
1153.csv	0.6	0.001	2	10	0.1	0.8
1154.csv	0.6	0.001	2	10	0.1	0.8
1155.csv	0.6	0.001	2	10	0.1	0.8
1156.csv	0.6	0.001	2	10	0.1	0.8
1157.csv	0.4	0.001	2	10	0.1	0.8
1158.csv	0.4	0.001	2	10	0.1	0.8
1159.csv	0.4	0.001	2	10	0.1	0.8
1160.csv	0.4	0.001	2	10	0.1	0.8
1161.csv	0.4	0.001	2	10	0.1	0.8
1162.csv	0.4	0.001	2	10	0.1	0.8
1163.csv	0.4	0.001	2	10	0.1	0.8
1164.csv	0.4	0.001	2	10	0.1	0.8
1165.csv	0.4	0.001	2	10	0.1	0.8
1166.csv	0.4	0.001	2	12	0.083333	0.8
1167.csv	0.4	0.001	2	12	0.083333	0.8
1168.csv	0.4	0.001	2	12	0.083333	0.8
1169.csv	0.4	0.001	2	12	0.083333	0.8
1170.csv	0.4	0.001	2	12	0.083333	0.8
1171.csv	0.4	0.001	2	12	0.083333	0.8
1172.csv	0.4	0.001	2	12	0.083333	0.8
1173.csv	0.4	0.001	2	12	0.083333	0.8
1174.csv	0.4	0.001	2	12	0.083333	0.8
1175.csv	0.4	0.001	2	12	0.083333	0.7
1176.csv	0.4	0.001	2	12	0.083333	0.7
1177.csv	0.4	0.001	2	12	0.083333	0.7

1178.csv	0.4	0.001	2	12	0.083333	0.7
1179.csv	0.4	0.001	2	10	0.1	0.7
1180.csv	0.4	0.001	2	10	0.1	0.7
1181.csv	0.4	0.001	2	10	0.1	0.7
1182.csv	0.4	0.001	2	10	0.1	0.7
1183.csv	0.4	0.001	2	10	0.1	0.7
1184.csv	0.4	0.001	2	10	0.1	0.7
1185.csv	0.4	0.001	2	10	0.1	0.4
1186.csv	0.4	0.001	2	10	0.1	0.4
1187.csv	0.4	0.001	2	10	0.1	0.4
1188.csv	0.4	0.001	2	10	0.1	0.4
1189.csv	0.4	0.001	2	10	0.1	0.4
1190.csv	0.4	0.001	2	10	0.1	0.4
1191.csv	0.4	0.001	2	10	0.1	0.4
1192.csv	0.4	0.001	2	10	0.1	0.4
1193.csv	0.4	0.001	2	10	0.1	0.4
1194.csv	0.4	0.001	2	10	0.1	0.4
1195.csv	0.4	0.001	2	10	0.1	0.4
1196.csv	0.4	0.001	2	10	0.1	0.4
1197.csv	0.3	0.001	2	9	0.111111	0.4
1198.csv	0.3	0.001	2	9	0.111111	0.4
1199.csv	0.3	0.001	2	9	0.111111	0.4
1200.csv	0.3	0.001	2	9	0.111111	0.4
1201.csv	0.3	0.001	2	9	0.111111	0.5
1202.csv	0.3	0.001	2	9	0.111111	0.5
1203.csv	0.3	0.001	2	9	0.111111	0.5
1204.csv	0.3	0.001	2	9	0.111111	0.5
1205.csv	0.3	0.001	2	8	0.125	0.5
1206.csv	0.3	0.001	2	8	0.125	0.5
1207.csv	0.3	0.001	2	8	0.125	0.5
1208.csv	0.25	0.001	2	8	0.125	0.5
1209.csv	0.25	0.001	2	8	0.125	0.5
1210.csv	0.25	0.001	2	8	0.125	0.5

8.2 Implementation of Model (Code)

CODE

Version 1:
Driver

```
package version1;

public class Driver {
     public static void main(String[] args) {
          Village v = new Village();
          v.modelInteractions();

     }

}
```

Person

```
package version1;

import java.util.ArrayList;

public class Person {

     public static double IMMUNITY_RATE = .001;   //
     immunity rate for the entire population
     public static double LATENT_PERIOD = 2; // time before
     a victim displays symptoms
     public static double INFECTED_PERIOD = 8;    //
     average duration of disease while symptoms are
     apparent, after which a victim either recovers or dies
     public static double REMOVAL_RATE = 1/INFECTED_PERIOD;
          // once infected period is over, the chance that
     a victim will recover and become immune
     public static double AVOIDANCE = .5;    // equal to 1-
     (avoidance rate)
     // the percent of interactions that result in the
     transmission of the infection when the transmitter is
     outwardly displaying symptoms as a result of being
     avoided by susceptible persons


     private double symptomSeverity;    // severity of
     symptoms outwardly displayed to others (0/1)
     private double age; // included but not used in this
     version of the model
     private boolean isDead;  // whether someone is alive,
     or was killed by the disease
     private boolean isInfected;   // true when a person is
     infected by the disease
     private boolean isImmune;     // whether or not a
     person is immune to the disease, either naturally or
     by recovering
```

```
private int sex;    // 0=female, 1=male (included, but
not used in this version of the model)

private double elapsedLat,elapsedInf,latTime,infTime;

public Person (boolean infect) {
     isDead = false;
     isInfected = infect;
     isImmune = false;
     symptomSeverity = 0;
     sex = (int)(Math.random()*2); // equal chance of
males and females in the population
     elapsedLat = 0;
     elapsedInf = 0;
     if (infect) {
          elapsedInf =
(int)(Math.random()*INFECTED_PERIOD);
     }
     latTime = LATENT_PERIOD + (int)(Math.random()*3 -
1);
     infTime = INFECTED_PERIOD + (int)(Math.random()*3
- 1);

     if (Math.random() <= IMMUNITY_RATE) {
          isImmune = true;
     }
     double temp = Math.random();  // used in
determining the person's age, based on percentage of
population that falls into certain age groups
     if (temp < .262)
          age = (int)(Math.random()*14+1);   //
randomly generated age: 1-14
     if (temp >= .262 && temp < .433)
          age = (int)(Math.random()*10+15);  //
randomly generated age: 15-24
     if (temp >= .433 && temp < .856)
          age = (int)(Math.random()*30+25);  //
randomly generated age: 25-54
     if (temp >= .856 && temp < .935)
          age = (int)(Math.random()*10+55);  //
randomly generated age: 55-64
     if (temp >= .935)
          age = (int)(Math.random()*20+65);  //
randomly generated age: 65-84 (upper bound is slightly
greater than life expectancy)
}
```

```
public void interact(ArrayList<Person> pop, double ip,
int pos) {

     if (isDead || isImmune || !isInfected)  // if the
person is either dead, immune, or healthy, then their
interactions are of no consequence to the model
          return;

     if (elapsedLat < latTime-1) { // if the victim is
out of the latency stage, they begin to show symptoms
          elapsedLat++;
     }
     else
          symptomSeverity = 1;

     if (elapsedInf < infTime-1) { // if the victim is
out of the infected stage, they either pass into in
immune state or are killed by the disease
          elapsedInf++;
     }
     else if (!isImmune) {
          isInfected = false;
          if (Math.random() < REMOVAL_RATE)  // chance
that the victim recovers from the disease and gains
immunity
               isImmune = true;
          else
               isDead = true;
     }

     for (int i=0; i<pop.size(); i++) {
          if (i != pos) {     // a person does not
interact with himself
               Person interactee = pop.get(i);
               if (!interactee.isDead() &&
!interactee.isImmune() && !interactee.isInfected()) {
                    if (symptomSeverity == 1)     //
if the infected person is showing symptoms, they are
less likely to transmit the disease to someone else,
by a factor of AVOIDANCE
                         ip = ip * AVOIDANCE;
                    if (Math.random() <= ip) {    //
probability that an infected person will infect a
susceptible person
                         interactee.infect();
                    }
               }
```

```
                    }
               }
          }

          public boolean isInfected() {
               return isInfected;
          }

          public boolean isDead() {
               return isDead;
          }

          public boolean isImmune() {
               return isImmune;
          }

          public void infect() {
               if (!isImmune)
                    isInfected = true;
          }

          public String toString() {
               if (isDead)
                    return "dead";
               if (isImmune)
                    return "immune";
               if (isInfected)
                    return "infected";
               return "susceptible";

          }



}
```

Village

```
package version1;

import java.io.BufferedReader;
import java.io.FileReader;
import java.io.PrintWriter;
import java.util.ArrayList;

public class Village {
```

```
private static double TRANS_RATE = .25; // quantifies
the rate of transmissibility for the disease

private PrintWriter file;
private PrintWriter masterFile;
private BufferedReader reader;
private ArrayList<Person> pop = new
ArrayList<Person>();     // contains a Person object
for everyone in the village
private double area;     // area of village in sq km
private int day;    // days since the start of the
simulation
private int numInfected;

public Village() {
     String text = "";
     try {
          reader = new BufferedReader(new
FileReader("master_file.csv"));
          try {
             StringBuilder builder = new
StringBuilder();
             String line = reader.readLine();

             while (line != null) {
                 builder.append(line);

builder.append(System.lineSeparator());
                 line = reader.readLine();
             }
             text = builder.toString();
         } finally {
             reader.close();
         }
     } catch(Exception e) {System.out.println("master
file read error");}

     String temp = "";
     file = null;
     for (int i=1111; i<9999; i++) {
          if (!text.contains(""+i)) {
               temp = ""+i+".csv";
               break;
          }
     }
     try {
          file = new PrintWriter(temp);
     } catch (Exception e) {System.out.println("file
writing error");}
     try {
          masterFile = new
PrintWriter("master_file.csv");
     } catch (Exception e) {System.out.println("master
file write error");}

     masterFile.print(text);
     masterFile.println(temp + "," + TRANS_RATE + ","
+ Person.IMMUNITY_RATE + "," + Person.LATENT_PERIOD +
"," + Person.INFECTED_PERIOD + "," +
Person.REMOVAL_RATE + "," + Person.AVOIDANCE);

     numInfected = 450;  // defines the original
number of infected people as between the range of 125-
149
     for (int i=0; i<numInfected; i++) {     //
     populates the population array with people that
are already infected
          pop.add(new Person(true));
     }
     for (int i=numInfected; i<1000; i++) {  //
     populates the population array with people that
are healthy
          pop.add(new Person(false));
     }
     area = 2.5;    // based on the population density
of Indonesia and the number of people in the village
     masterFile.close();
}

public void modelInteractions() {

     day = 0;
     int healthy = 0;
     int dead = 0;
     int infected = numInfected;   // stores the
number of people who are infected, so that the loop
runs correctly
     int immune = 0;
     int susceptible = 0;


     file.println("DAY,HEALTHY,DEAD,INFECTED,IMMUNE,SU
SCEPTIBLE");
```

```
     while (infected>0 && day<5000) {   // the model
continues to run until the number of infected people
reaches 0;
          day++;
          healthy = 0;   // number of people w/out the
disease
          dead = 0;
          infected = 0;  // number of people who have
the disease, including those without symptoms
          immune = 0;    // number of people who are
immune to the disease, either passively or after
recovering

          for (Person p: pop) {    // counts the
number of people who are healthy, infected, and dead
at the beginning of each day
               if (p.isDead())
                    dead++;
               if (!p.isDead() && !p.isInfected())
                    healthy++;
               if (p.isInfected())
                    infected++;
               if (p.isImmune())
                    immune++;
          }
          susceptible = healthy - immune; // number of
people who can still be infected with the disease

          double temp = (1-(2.5*TRANS_RATE*(infected-
infected*infected/(healthy+infected))/area)/susceptibl
e);
          double interactionPossibility = 1-
nthRoot(temp,infected);
               // percent chance that an infected
person will interact with a susceptible person and
will transmit the disease to them
               // calculation is based on a formula we
developed, and is documented in our paper

          for (int i=0; i<pop.size(); i++) { // calls
the interact() method for all of the people in the
population
               pop.get(i).interact(pop,
interactionPossibility, i);
          }
```

```
                file.println(day + "," + healthy + "," +
        dead + "," + infected + "," + immune + "," +
        susceptible);
            }
            file.close();
        }

        public double nthRoot(double val, int root) {
            double ans = 0.00001;
            while(Math.pow(ans,root)<val) {
                ans+=.01;
            }
            ans-=.01;
            while(Math.pow(ans,root)<val) {
                ans+=.001;
            }
            ans-=.001;
            while(Math.pow(ans,root)<val) {
                ans+=.0001;
            }
            ans-=.0001;
            while(Math.pow(ans,root)<val) {
                ans+=.00001;
            }
            ans-=.00001;
            while(Math.pow(ans,root)<val) {
                ans+=.000001;
            }
            ans-=.000001;
            return ans;
        }
}
```

Analysis:

Severity Indicator

```
package analysis;

import java.io.BufferedReader;
import java.io.FileReader;
import java.util.Scanner;
import java.util.StringTokenizer;

public class SeverityIndicator {

        private static BufferedReader reader;
```

```
public static void main(String[] args) {

     Scanner scan = new Scanner(System.in);
     System.out.println("Enter file id number:");
     String fileId = scan.next();
     String text = "";

     reader = null;
     try {
          reader = new BufferedReader(new
FileReader(fileId + ".csv"));
          try {
             StringBuilder builder = new
StringBuilder();
             String line = reader.readLine();

             while (line != null) {
                 builder.append(line);

builder.append(System.lineSeparator());
                 line = reader.readLine();
             }
             text = builder.toString();
         } finally {
             reader.close();
         }
     } catch (Exception e) {};

     text = text.replace("\n", ",");
     StringTokenizer tok = new
StringTokenizer(text,",");
     int[][] vals = new int[3000][6];

     int row = 0;
     for (int i=0; i<6; i++) {     // loops out of the
file headers
          tok.nextToken();
     }

     while (tok.hasMoreTokens()) {
          for (int i=0; i<6; i++)
               vals[row][i] =
Integer.parseInt(tok.nextToken().trim());
          row++;
     }
     int maxRow = 0;
     StringTokenizer tok2 = new
StringTokenizer(text,",");
     for (int i=0; i<6; i++) {     // loops out of the
file headers
          tok2.nextToken();
     }

     for (int i=0; i<3000; i++) {
          if (vals[i][0] > maxRow)
               maxRow = vals[i][0];
     }

     int[][] temp = new int[maxRow][6]; // trims array
to only include the data from the excel sheet
     for (int i=0; i<maxRow; i++)
          for (int j=0; j<6; j++)
               temp[i][j] = vals[i][j];
     vals = temp;

     int M = 0;
     for (int i=0; i<vals.length; i++) {
          if (vals[i][3] > M)
               M = vals[i][3];     // maximum number
of infections at any time
     }
     int n = vals[0][1] + vals[0][2] + vals[0][3];
     // population at day 0
     int D = vals[vals.length-1][2];    // number of
deaths on the last day
     int h = 999;
     int iLoc = 1;  // time in days when rate of new
infections is at a minimum
     for (int i=0; i<vals.length-1; i++) {
          if (vals[i+1][3]-vals[i][3] < h) {
               h = vals[i+1][3]-vals[i][3];
               iLoc = i+1;
          }
     }

     double zeta = ((10*Math.pow(M,2)*D*iLoc));
     zeta = zeta/(Math.pow(n,3)*vals.length);
     zeta = Math.round(zeta);
     System.out.println("Calculated zeta value:
"+zeta);

}
```

```
}
```

Version 2:

Driver

```
package version2;

import version1.Village;

public class Driver {

     public static void main(String[] args) {
          Village v = new Village();
          v.modelInteractions();

     }

}
```

Person

```
package version2;

import java.util.ArrayList;

public class Person {

     public static double IMMUNITY_RATE = .001;   //
     immunity rate for the entire population
     public static double LATENT_PERIOD = 2; // time before
     a victim displays symptoms
     public static double INFECTED_PERIOD = 10;   //
     average duration of disease while symptoms are
     apparent, after which a victim either recovers or dies
     public static double REMOVAL_RATE = 1/INFECTED_PERIOD;
          // once infected period is over, the chance that
     a victim will recover and become immune
     public static double AVOIDANCE = .7;    // equal to 1-
     (avoidance rate)
     // the percent of interactions that result in the
     transmission of the infection when the transmitter is
     outwardly displaying symptoms as a result of being
     avoided by susceptible persons
```

```
private double symptomSeverity;    // severity of
symptoms outwardly displayed to others (0/1)
private double age; // included but not used in this
version of the model
private boolean isDead;  // whether someone is alive,
or was killed by the disease
private boolean isInfected;   // true when a person is
infected by the disease
private boolean isImmune;     // whether or not a
person is immune to the disease, either naturally or
by recovering
private int sex;    // 0=female, 1=male (included, but
not used in this version of the model)

private double elapsedLat,elapsedInf,latTime,infTime;

public Person (boolean infect) {
     isDead = false;
     isInfected = infect;
     isImmune = false;
     symptomSeverity = 0;
     sex = (int)(Math.random()*2); // equal chance of
males and females in the population
     elapsedLat = 0;
     elapsedInf = 0;
     if (infect) {
          elapsedInf =
(int)(Math.random()*INFECTED_PERIOD);
     }
     latTime = LATENT_PERIOD + (int)(Math.random()*3 -
1);
     infTime = INFECTED_PERIOD + (int)(Math.random()*3
- 1);

     if (Math.random() <= IMMUNITY_RATE) {
          isImmune = true;
     }
     double temp = Math.random();  // used in
determining the person's age, based on percentage of
population that falls into certain age groups
     if (temp < .262)
          age = (int)(Math.random()*14+1);   //
randomly generated age: 1-14
     if (temp >= .262 && temp < .433)
          age = (int)(Math.random()*10+15);  //
randomly generated age: 15-24
     if (temp >= .433 && temp < .856)
```

```
          age = (int)(Math.random()*30+25);  //
randomly generated age: 25-54
     if (temp >= .856 && temp < .935)
          age = (int)(Math.random()*10+55);  //
randomly generated age: 55-64
     if (temp >= .935)
          age = (int)(Math.random()*20+65);  //
randomly generated age: 65-84 (upper bound is slightly
greater than life expectancy)
}

public void interact(ArrayList<Person> pop, double ip,
int pos) {

     if (isDead || isImmune || !isInfected)  // if the
person is either dead, immune, or healthy, then their
interactions are of no consequence to the model
          return;

     if (elapsedLat < latTime-1) { // if the victim is
out of the latency stage, they begin to show symptoms
          elapsedLat++;
     }
     else
          symptomSeverity = 1;

     if (elapsedInf < infTime-1) { // if the victim is
out of the infected stage, they either pass into in
immune state or are killed by the disease
          elapsedInf++;
     }
     else if (!isImmune) {
          isInfected = false;
          if (Math.random() < REMOVAL_RATE) {     //
chance that the victim recovers from the disease and
gains immunity
               isImmune = true;
               if ((age <= 14 || age >= 55) &&
Math.random() < .75) {   // if person is either a
child or elderly, then they have a 75% less chance of
surviving when compared to any other person
                    isImmune = false;
               }
          }
          else
               isDead = true;
     }
     for (int i=0; i<pop.size(); i++) {
          if (i != pos) {     // a person does not
interact with himself
               Person interactee = pop.get(i);
               if (!interactee.isDead() &&
!interactee.isImmune() && !interactee.isInfected()) {
                    if (symptomSeverity == 1)     //
if the infected person is showing symptoms, they are
less likely to transmit the disease to someone else,
by a factor of AVOIDANCE
                         ip = ip * AVOIDANCE;
                    if (Math.random() <= ip) {    //
probability that an infected person will infect a
susceptible person
                         interactee.infect();
                    }
               }
          }
     }
}

public boolean isInfected() {
     return isInfected;
}

public boolean isDead() {
     return isDead;
}

public boolean isImmune() {
     return isImmune;
}

public void infect() {
     if (!isImmune)
          isInfected = true;
}

public String toString() {
     if (isDead)
          return "dead";
     if (isImmune)
          return "immune";
     if (isInfected)
          return "infected";
     return "susceptible";
     }



}
```

Village

```
package version2;

import java.io.BufferedReader;
import java.io.FileReader;
import java.io.PrintWriter;
import java.util.ArrayList;

public class Village {

     private static double TRANS_RATE = .053954;  //
     quantifies the rate of transmissibility for the
     disease, .053954 is based on the transmissability of
     ebola

and other diseases spread through bodily fluids
     private static double area = 2.5;  // area of village
     in sq km
     private static int initPop = 300;
     private static int initInfect = 145;

     private PrintWriter file;
     private PrintWriter masterFile;
     private BufferedReader reader;
     private ArrayList<Person> pop = new
     ArrayList<Person>();     // contains a Person object
     for everyone in the village
     private int day;    // days since the start of the
     simulation
     private int numInfected;

     public Village() {
          String text = "";
          try {
               reader = new BufferedReader(new
     FileReader("master_file2.csv"));
               try {
                  StringBuilder builder = new
     StringBuilder();
```

```
                  String line = reader.readLine();

                  while (line != null) {
                      builder.append(line);

     builder.append(System.lineSeparator());
                      line = reader.readLine();
                  }
                  text = builder.toString();
              } finally {
                  reader.close();
              }
          } catch(Exception e) {System.out.println("master
     file read error");}

          String temp = "";
          file = null;
          for (int i=1111; i<9999; i++) {
               if (!text.contains(""+i)) {
                    temp = ""+i+"v2.csv";
                    break;
               }
          }
          try {
               file = new PrintWriter(temp);
          } catch (Exception e) {System.out.println("file
     writing error");}
          try {
               masterFile = new
     PrintWriter("master_file2.csv");
          } catch (Exception e) {System.out.println("master
     file write error");}

          masterFile.print(text);
          masterFile.println(temp + "," + TRANS_RATE + ","
     + Person.IMMUNITY_RATE + "," + Person.LATENT_PERIOD +
     "," + Person.INFECTED_PERIOD + "," +

Person.REMOVAL_RATE + "," + Person.AVOIDANCE);

          numInfected = initInfect;     // defines the
     original number of infected people as between the
     range of 125-149
          for (int i=0; i<numInfected; i++) {     //
          populates the population array with people that
     are already infected
               pop.add(new Person(true));
          }
          for (int i=numInfected; i<initPop; i++) {    //
          populates the population array with people that
     are healthy
               pop.add(new Person(false));
          }
          masterFile.close();
     }

     public void modelInteractions() {

          day = 0;
          int healthy = 0;
          int dead = 0;
          int infected = numInfected;   // stores the
     number of people who are infected, so that the loop
     runs correctly
          int immune = 0;
          int susceptible = 0;


          System.out.println("DAY,HEALTHY,DEAD,INFECTED,IMM
     UNE,SUSCEPTIBLE");

          while (infected>0 && day<3000) {   // the model
     continues to run until the number of infected people
     reaches 0;
               day++;
               healthy = 0;   // number of people w/out the
     disease
               dead = 0;
               infected = 0;  // number of people who have
     the disease, including those without symptoms
               immune = 0;    // number of people who are
     immune to the disease, either passively or after
     recovering

               for (Person p: pop) {    // counts the
     number of people who are healthy, infected, and dead
     at the beginning of each day
                    if (p.isDead())
                         dead++;
                    if (!p.isDead() && !p.isInfected())
                         healthy++;
                    if (p.isInfected())
                         infected++;
                    if (p.isImmune())
```

```
                immune++;
            }
            susceptible = healthy - immune; // number of
people who can still be infected with the disease

            double temp = (1-(2.5*TRANS_RATE*(infected-
infected*infected/(healthy+infected))/area)/susceptibl
e);
            double interactionPossibility = 1-
nthRoot(temp,infected);
                // percent chance that an infected
person will interact with a susceptible person and
will transmit the disease to them
                // calculation is based on a formula we
developed, and is documented in our paper

            for (int i=0; i<pop.size(); i++) { // calls
the interact() method for all of the people in the
population
                pop.get(i).interact(pop,
interactionPossibility, i);
            }
            System.out.println(day + "," + healthy + ","
+ dead + "," + infected + "," + immune + "," +
susceptible);
        }
        file.close();
}

public double nthRoot(double val, int root) {
        double ans = 0.00001;
        while(Math.pow(ans,root)<val) {
            ans+=.01;
        }
        ans-=.01;
        while(Math.pow(ans,root)<val) {
            ans+=.001;
        }
        ans-=.001;
        while(Math.pow(ans,root)<val) {
            ans+=.0001;
        }
        ans-=.0001;
        while(Math.pow(ans,root)<val) {
            ans+=.00001;
        }
        ans-=.00001;
        while(Math.pow(ans,root)<val) {
            ans+=.000001;
        }
        ans-=.000001;
        return ans;
    }
}
```