Spring 2019

Fitting the SIR Epidemiological Model to Influenza Data

Madeline Dorr
University of Maine

FITTING THE SIR EPIDEMIOLOGICAL MODEL TO INFLUENZA DATA

by

Madeline Dorr

A Thesis Submitted in Partial Fulfillment
of the Requirements for a Degree with Honors
(Mathematics)

The Honors College

University of Maine

May 2019

Advisory Committee:
David Hiebeler, Professor of Mathematics, Advisor
Edie Elwood, Adjunct Assistant Professor in Sociology and Preceptor in the Honors College
Andre Khalil, Associate Professor of Chemical and Biomedical Engineering
Peter Stechlinski, Assistant Professor of Mathematics
Zheng Wei, Assistant Professor of Statistics

ABSTRACT

This project sought to provide thorough instructions to fitting the SIR epidemiological model to influenza data and defend its use in this context. Directions for coding the SIR model in the R programming language are detailed. This includes estimating parameter values, such as infection and recovery rate, and how to double check these values. This project also included analysis of problems that can arise when fitting this model. This includes accounting for vaccination rate and issues with the nature of this type of data. Either these problems were explored, and solutions were provided, or suggestions were provided for future research.

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

I. INTRODUCTION

There are many ways to mathematically model disease surveillance data. These models are important to predict the disease behavior and to optimize preventative measures, such as vaccinations and general cleanliness. For example, analyzing data about influenza epidemics shows that the season typically starts around May in the American Tropics.[1] This information can then be used to start vaccinations in April, and then modelling this next season's data can further refine the timing of preventative measures.[1] Using this to more efficiently allocate resources can decrease morbidity and mortality of the influenza.[2] It is also important to note that while the mortality rate of influenza is low in Northern America for the typical flu season,[2] there are still socio-economics effects on the population that make this worthwhile to study.[3] These effects include the cost of medical care and loss of productivity. It has been estimated that the economic cost because of the influenza is between 13.9 thousand to 957.9 million dollars per season in the US as of 2010.[3] In short, using modeling and analysis of flu data to implement strategic prevention measures is important because it can lead to decreases in mortality rates and lessen the socio-economic impact that the influenza has on the nation.

One such mathematical model that can be used to study influenza data is the deterministic SIR epidemiological model. In this model, the population is divided into three separate groups, or compartments, that describe the group's status, relevant to the disease in question, at a point in time.[4] These groups are Susceptible (S), Infectious (I), and Recovered (R). People within the S group have not yet been infected by the disease of interest. The I group consists of people who are currently infected with the disease and capable of infecting others, while people in the R group were formerly infected and have since recovered from the disease (and as a result are immunized). **Equation** 1 shows the system of ordinary differential equations used to determine how much of the population is within each group at a specific time (t) for the SIR model.

Equation 1.

$$
\boxed{
\begin{aligned}
&N = S + I + R \\
&\frac{dS(t)}{dt} = -\beta * \frac{S*I}{N} \\
&\frac{dI(t)}{dt} = \beta * \frac{S*I}{N} - \gamma*I \\
&\frac{dR(t)}{dt} = \gamma*I
\end{aligned}
}
$$

Where N is the population size, β is the infection (or contact) rate and γ is the recovery rate.[4] The SIR model assumes that there is a constant population size N, that the rates of infection and recovery remain constant, and that it is a well-mixed population, meaning that there is a chance for any infectious individual to contact and contaminate any susceptible individual.[5] There have been previous papers and studies using the SIR model to analyze influenza data that support its application to the data sets used later in this paper.[6,7,8] However, they have not been applied to model the influenza in the US.

II. DATA

The datasets used in this paper are available for public use through the Center for Disease Control (CDC) website.[9] As this paper exclusively does secondary analysis of existing de-identified and publicly available data, IRB approval was not necessary. This IRB exemption is in accordance with Federal Policy 45 CFR 46.102[10] and the University of Maine Policy Concerning the Protection of Human Subjects of Research.[11]

The information was collected by the CDC from public health and clinical laboratories across the nation. The data was collected by flu season and grouped into weeks, starting at week 40 of the year and continuing for 52 total weeks. The number of total specimens tested and the number of infected specimens by strain are reported for each week. Data is available for flu seasons from 1997 until the present.

In terms of the SIR model, the total number of infected specimens, regardless of strain, represent the Infectious (I) category of the model. However, the total number of specimens tested does not accurately represent the Susceptible (S) category of the model. This is a result of data collection from different regions of the nation using different testing practices, resulting in varying amounts of specimens tested as well as seemingly varying rates of infection.[9] This is also a result of under-reporting that occurs with the influenza virus.[2] Therefore the population size needs to be estimated. The method to estimate population size used in this paper will be addressed further in Chapter III.

III: CODING THE SIR MODEL IN R

Before analyzing any data, at least a simplistic version of the code must first be set up. For the first round of code for the SIR model, the different compartmental groups are set up as proportions. This is a small adjustment to the **sir.model()** code found in the Appendix.

```
##Proportion of Susceptible Individuals at Start

p.total.s <- (total.s[1] / (total.s[1]+total.i[1]))

##Proportion of Infectious Individuals at Start

p.total.i <- (total.i[1] / (total.s[1]+total.i[1]))

init.v <- c(S=p.total.s, I=p.total.i, R=0)
```

This method provides cleaner and usually more easily interpretable results. The other method is to have the compartmental groups as the actual numerical values of individuals per group. Since the SIR model assumes there is a fixed population size N, both methods produce the same results. The proportional method simply makes it easier to quickly identify the relationship between the groups at any given point. This relationship is still present in the non-normalized version, but slightly obscured.

However, for multiple reasons, I quickly shifted from using the proportions to using the non-normalized method. For starters, it fit better with the data sets I had. The data set provided the number of individuals per week within the S and I groups, so using proportions required transforming the data before fitting it to the model. Another problem the proportional method had was that N came from adding the initial S and I group values. This is a problem specific to flu data because it would be abnormal for seemingly healthy individuals to seek medical care, so not all of the individuals in the S compartment would have visited a clinic. This means that the true N, calculated by the number of individuals within all three compartments at a given time, is not represented within the data. Since the N value is not represented within the data, N is included within the parameter vector in the **sir.model()** code along with the β and γ values.

Another motivation to estimate N is to find the effective population size. There is not realistically total mixing in the population as not every infectious individual has at least a chance of contacting a susceptible individual. However, the SIR model works off the assumption that the population is well-mixed in this context. There is an effective population size where the population is well-mixed that can be estimated. This effective population size is the N that should be used when running the SIR code because this is the population estimate that will allow for the data to be fitted.

The **sir.model()** code initially has guesses for the values of β, γ, and N in the parameter vector. This is done so that the code will actually run, which allows for forward progress as well as some basic debugging. As is, this code produces predictions of the values of S, I, and R per each week of the flu season, as well as a graph showcasing the numerical results of the ordinary differential equations (ODEs) of S, I, and R. The graph in Figure 1 clearly shows that the initial guess for the parameter values must be incorrect. Comparing the green line, representing the numerical results of the ODEs for the I group, and the black line, the actual data, shows that the parameter values must be better estimated. Ideally, the green and black line would very closely match, as this would indicate that the model fits the data.

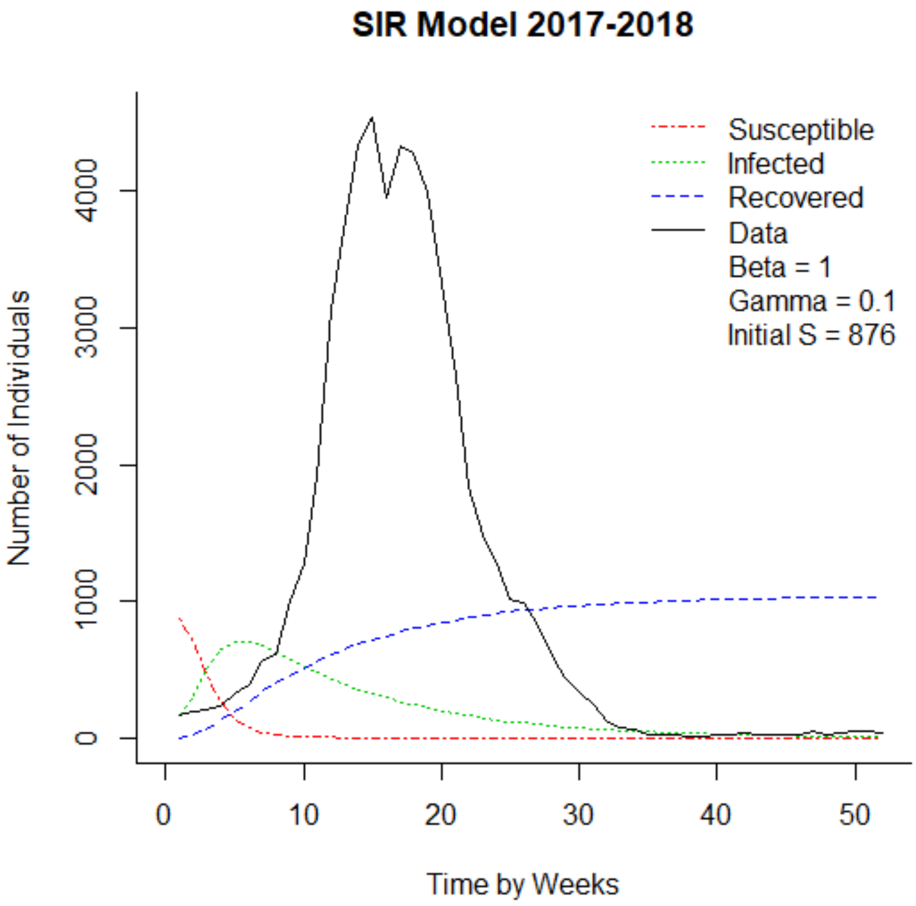

Figure 1. This graph shows the SIR model results with the initial guesses of β and γ. The β and γ values used are found in the legend, where β is 1 and γ is 0.1. The initial S value used is also found in the legend, where S=876.

With the SIR model code set up, better estimates of the values of the parameters are needed. To find these, code is set up to run the **sir.model()** function with a range of β, γ, and N values. The initial guesses made for these parameters will first be run through the **L1()** function, which calls the **sir.model()** code and calculates the sum of squared residuals (SSR) between values from the data set and the predicted values when using the current β and γ values. Each different combination of the parameter values within a given range are run through the **L1()** function, and all of them have their SSR calculated.

The SSR is a method of comparing different models (in this case specifically different parameter values) to mathematically determine the best fitting model for the data. The model produces numerical results of the ODEs for S, I, and R over time. For our purposes, the I group is the main point of interest because this is the group where people are actively sick. Therefore, the residuals of the I group are calculated. To calculate the residuals, the predicted I values must be determined by the model using the numerical results of the ODEs. The residuals of a model are the difference between the predicted I values and the I values from the data set at each given time point. The SSR is found by adding together all the residual values and then squaring them. This formula is shown in **Equation 2**. Relatively smaller residuals indicate better fitting models because that would mean the model more accurately predicts the data values. The larger the absolute value of a residual, the more inaccurate the model. Since SSR is calculated from summation of the residuals, the model with the smallest SSR is the best fitting model for the data over the entire time period.

Equation 2.

$$\mathbf{L(\Theta) = \sum_{i=1}^{n=52} [y(t_i) - \hat{y}(t_i)]^2}, \quad i=1, 2, \ldots, 52$$

Where $\Theta = (\beta, \gamma, N)$, or the parameters of the SIR model,

$y(t_i)$ are the observations from the data at time t_i, and

$\hat{y}(t_i) = f(t_i, \Theta)$, and are the predictions from the ODEs at time t_i.

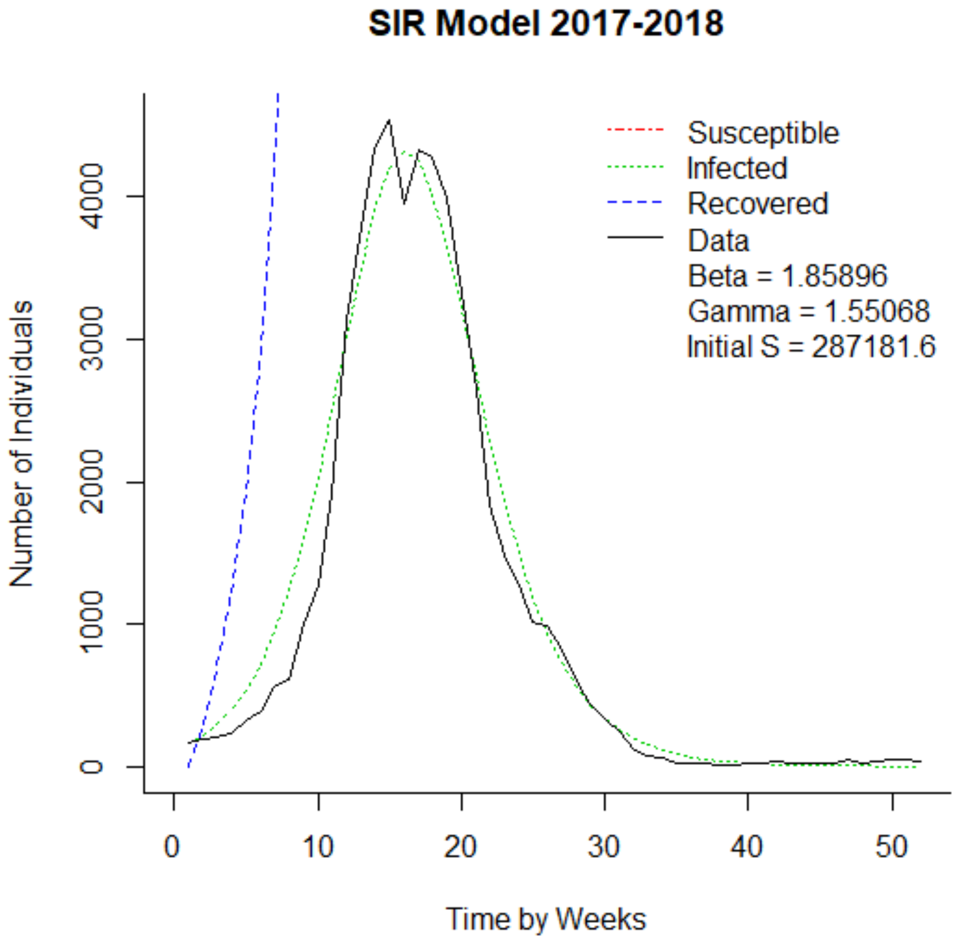

Figure 2. This graph shows the resulting SIR model when the optimized parameter values are used. The β, γ, and initial S value used are found in the legend in that respective order.

Once the **L1()** function has returned optimized values for the parameters based off the model SSR, graphing the model again helps to visualize if the code is running well so far. The Figure 2 graph is produced by plugging the estimates β=1.86, γ=1.55, and N=287181 from the **L1()** function into the **sir.model()** code, instead of the initial guesses for the parameter values. It's function is highlighted in **Equation** 1. The graph clearly shows that these new parameter values make the model fit the data much better than before. However, it is not guaranteed that these are the absolute best estimates for the parameter values. This will be explored further in the next chapter.

IV. HEAT MAP

It is possible that the **L1()** function may not perform as expected if the SSR values behave irregularly. For example, there may be a local minimum for the SSR values that is not a global minimum. In this case, the function could inaccurately conclude what the best estimate is for the parameter value. A local minimum would trick this function because it would produce a smaller SSR value than any combination of parameter values within a close range. The function continues to check new parameter values only so long as the SSR values keep decreasing from estimate to estimate. It is almost like water flowing down a hill. In the most common situation, gravity pulls the water down to the very bottom of the hill (the smallest SSR). However, it may be the case that the hill is not nicely conical like most other hills but is instead misshapen with rocks and lumps. These rocks and lumps could create divots that the water collects in and forms a puddle, stopped from reaching the bottom of the hill.

In order to determine that there are no local minimums or irregularities that affect our parameter estimates, a **heat map** (in the Appendix) is created. This calculates the SSR over a range of parameter values, for each combination within that range. This serves to visually represent the shape of the metaphorical hill.

The heat map graph, shown in Figure 3, visually depicts which values of β and γ produce a model with minimized SSR. The x-axis represents the β values and the y-axis represents the γ values. The smaller the SSR values are represented by the red color in the center. As the SSR increases, the graph changes color according to the rainbow. This means the red and orange areas are the better parameter estimates, while the blue and purple areas provide the worst model fit.

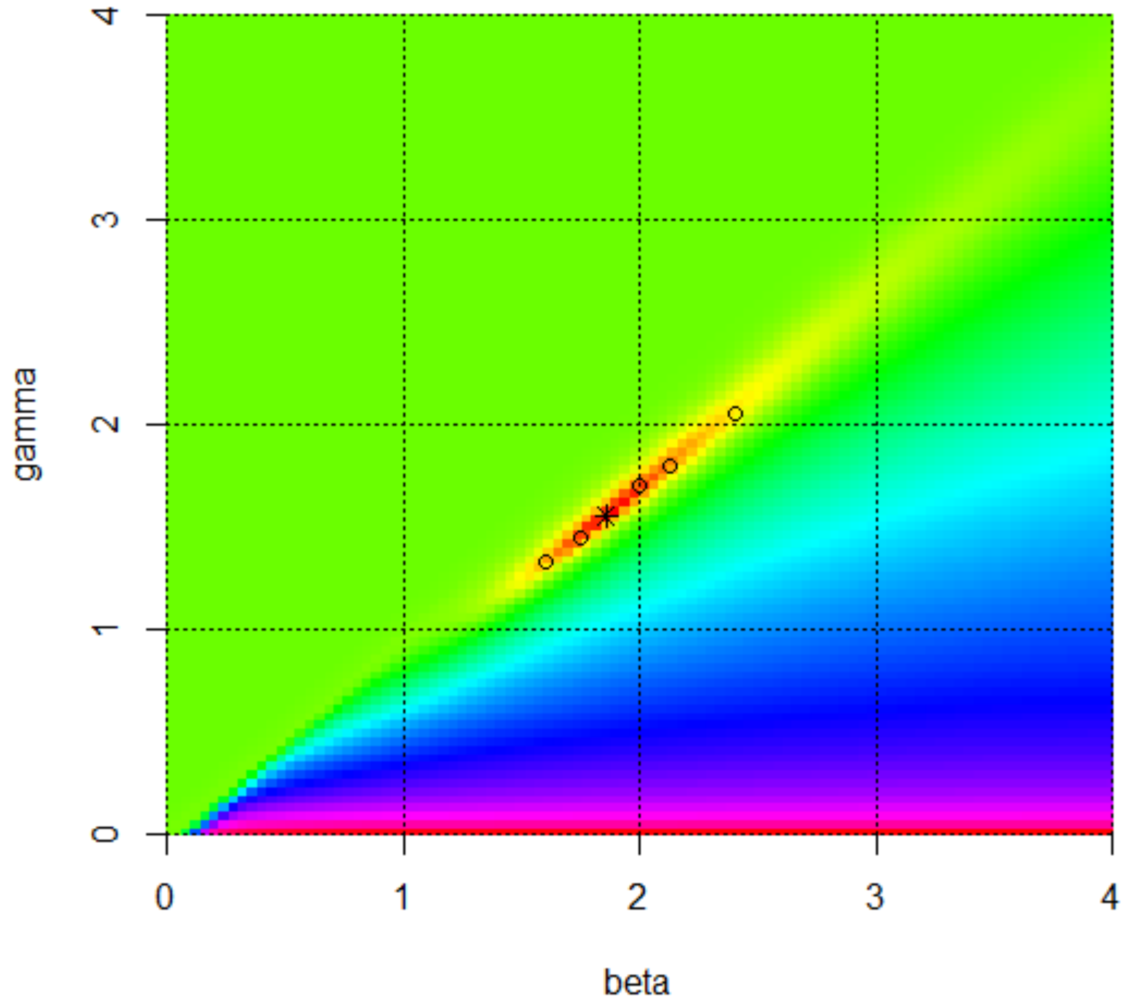

Figure 3. This is a heat map that visually shows the ranges of β and γ where the SSR are minimized. The red area shows the values where the SSR are smallest. The SSR values, shown by the colors, range from 4.0e+7 to 5.2e+7. The points added are the points of interest where the SSR appears to be minimized. The star is the point the optimization code chose as the best estimates for β and γ.

This heat map is used to find the absolute minimum SSR values instead of taking the derivatives with respect to the parameters. This is in part because there is not a specific analytic expression known for f(t, Θ), the function producing the SSR values. This is because f(t, Θ) is evaluated through calls to the numerical ODE solver. As such, the derivatives of f(t, ϴ) cannot be evaluated, so another method is needed for finding the absolute minimum of SSR, and the heat map is used instead.

Unfortunately, the heat map also indicates that there is potentially some irregularity in the shape of the metaphorical hill. Instead of the water collecting nicely into a small little puddle, there is a gash in the earth collecting the flowing water. This means that there is potentially a large range of values for β and γ where the SSR varies little or not at all. It is necessary to further explore the parameter values along this ravine in order to ensure the parameter estimates provided by the **L1()** function are actually the best estimates.

As indicated in the **heat map** code in the Appendix, the points along the ravine have been manually added in order to capture the nuanced elevation changes. In summary, the points are placed where the colors in the heat map indicate a change in the SSR value. This includes at the edges of the ravine and places inside the ravine where the red color deepens, so the SSR decreases. The point provided by the **L1()** code is also placed on the heat map.

The next step is to graph the SIR model with each of these different parameter values in order to see if there is any obvious visual difference. If the graphs for parameter values, produced from the points on the heat map, produce models that are clearly worse fits, these estimates can be disregarded.

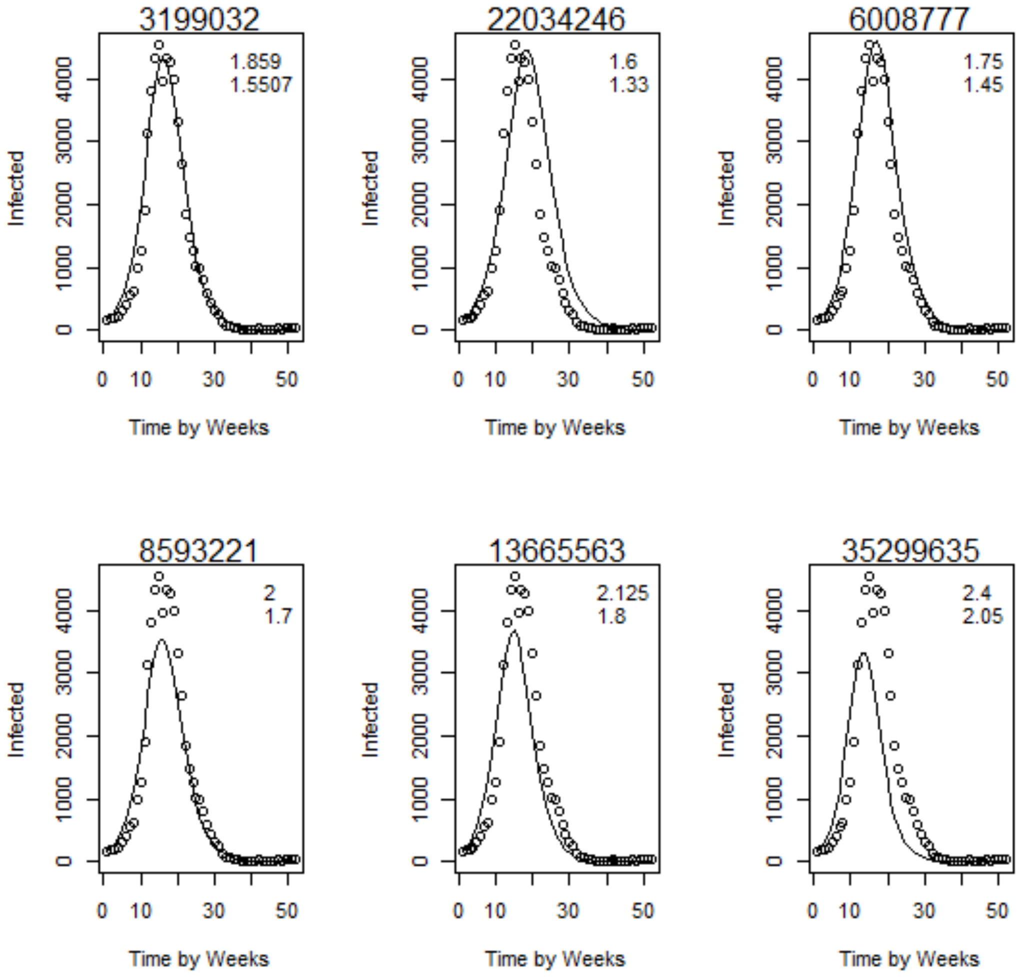

Figure 4. This shows six iterations of the same code with different parameter values. These are the parameter values that are the points of interest in Figure 3. The top left graph shows the iteration with the parameter values chosen by the code that optimizes the parameter estimates through minimizing the SSR. The y axis is the number of infected people. The first value in the legend is β, and the second value is γ. The SSR produced from these models are found at the top of each graph. The SSR values and close visual inspection show that the top left graph has the best fit for parameter values.

Figure 4 graphs the SIR models with different parameter estimates. Only the I group is shown in these graphs, as that is the group of the most interest. The line represents the model fit, while the circles are the actual data points. The numbers in the top right corner of each model graph are the β and γ values respectively.

The model on the top left of the image is produced using the parameter estimates provided by the **L1()** code. This image indicates that the bottom three models are clearly worse fits, and these parameter estimates can be ignored. The center top and right top models require a slightly closer look, but visual inspection still indicates that these parameter estimates do not provide the best model fit. Therefore, despite the potential irregularity shown on the heat map, the **L1()** function did indeed function as expected.

V. UPDATING CODE AND CHECKING RELIABILITY

Now that the population size N has been estimated the code needs to be tweaked so that N is no longer a parameter. Reducing the number of parameters in a model has many benefits. This reduces the computational cost of the model, allowing the code to run faster. Reducing computational cost also helps minimize the complexity of the problem. However, in this particular case it can also be used to reaffirm that the code is working correctly.

It can sometimes be difficult to determine if complex code is executing correctly. Removing N as a parameter will help determine the if the code produces reliable results. In this context, reliable means that the results are consistent. The accuracy of the model is determined using the SSR (described in detail in Chapter III). An unreliable model will give different results even when the initial conditions remain unchanged. This alone is a problem because the SIR model is deterministic, not stochastic, so there should not be variation in the results when the initial conditions are the same. Unreliable results can lead to wrong interpretations and bad predictions. Therefore, it is important to ensure the code produces reliable results.

The code is adjusted by making some changes in the initial condition and parameter vectors fed into the code. The complete changes between this version of the code and the code from Chapter III are highlighted in the **sir.model.vacc()** code found in the Appendix. The **sir.model.vacc()** shows four iterations of coding for the SIR model with different parameters. As labeled, two of these iterations use the coding method from

Chapter III, and the other two use the method discussed in this chapter to remove N as a parameter.

By removing N as a parameter and running the code again, the results of the two models can be compared to determine reliability. In theory, removing N as a parameter and instead using the estimate for N within the initial conditions should produce the same results. That is the case with this code. Taking any of the three parameters (N, β, or γ) and fixing them will still produce the same results. Therefore, the code is reliable and the SSR determines its accuracy. This indicates that the modeling method described in this paper is appropriate for modeling influenza data.

VI. ACCOUNTING FOR VACCINATIONS

An area of great interest in studying and modeling flu data is the relationship between vaccination and the amount of people infected. Using the CDC website where the data come from, there is also information on the percent of adults and children vaccinated in a given year. In this section, the flu data for 2012 to 2013 is modeled. For this particular year, 41.5% of adults and 56.6% of children (people 17 or younger) received the flu vaccine.[12]

Starting with the more conservative vaccination percentage, the **sir.model()** code is adjusted to account for 41.5% of the S group actually starting within the R group. This requires using both rounds of model code done in Chapter III and V. For the unvaccinated version of the model, the methods used in these chapters is followed exactly. However, for the model that accounts for vaccination, there are some adjustments that need to be made.

For the model accounting for vaccination, the initial parameter values must be changed. Since vaccination provides immunity to a disease, people with vaccinations will actually begin in the R group. The model is run so that the value of N as a parameter is estimated while vaccination rate is accounted for. This means the estimated value of N differs whether vaccination rate is included in the model or not. This step cannot be skipped because it is possible that the estimated population size will differ based on this adjustment to the S and R groups. In the Appendix, full coding is shown for setting up the models and comparing them under **sir.model.vacc**. These changes focus on the initial S, I, and R values used, and the **sir.model()** code can be adjusted using the following two lines of code:

```
init.v <- c(I=total.i[1],R=(41.5/(100-41.5))*total.s[1])

state.value <- c(I=total.i[1],R=(41.5/(100-41.5))*total.s[1])
```

Once this is done, the method in Chapter V can be used. This iteration of the model should be the version that accurately represents how accounting for vaccination changes the SIR model. Then, similar to the method explained in Chapter III, the residuals of the two models are compared in order to determine which model better fits the data.

Looking at the residuals shows that the models did not behave as initially expected. The SIR model that accounts for vaccinated individuals has a significantly larger residual, suggesting that this model is a worse fit than the model that does not account for vaccination. The **sir.model.vacc** code was run again using 56.6% vaccination rate, as that was the upper end of the vaccination range provided from the CDC website. Again, this results in the residuals indicating that the model accounting for vaccination is a poor fit compared to the other model.

These results are peculiar. A more likely result would be that there is no difference in residuals between the two models. If this occurred, it could be due to how the data was collected. Individuals who are vaccinated are not measured and differentiated from unvaccinated individuals. Lacking this variable in the data set would mean that the best β and γ values were chosen so that vaccination rate was implicitly accounted for with these values (meaning that the true β and γ values are different, but the values used best estimate the behavior of the data set). However, the residuals were not similar values, but instead the residual for the model accounting for vaccination had residuals that were nearly 100 times larger.

The discrepancy in the residuals prompted further exploration. As the vaccination rates from the CDC website did not appear to accurately capture the vaccination rate, a range of potential rates were considered. This involved inputting different vaccination rates into the code and observing if the discrepancy between the residuals decreased or increased. The difference between the residuals decreased as the proposed vaccination rate decreased. Eventually, a proposed vaccination rate for the model yielded a residual value smaller than the model that does not account for vaccination.

The residuals indicated that the behavior of the data is best captured with a 0.005% vaccination rate for the flu season over 2012 and 2013. This is illustrated in Figures 5 and 6. Figure 5 shows the model fit without accounting for vaccination, and Figure 6 shows the model fit when accounting for a vaccination rate of 0.005%. The residuals calculated are reported in these figures as well, and they are very similar numbers.

A possible explanation for this might stem from the set up of the model. It is possible that vaccination rate was accounted for when estimating N, with population size and vaccination rate acting as confounding variables. As such, the model cannot distinguish between N and vaccination rate, and they are both essentially represented by the parameter N. This would explain why a vaccination rate so close to zero would produce a relatively small SSR. A future model with ongoing vaccination rate, as opposed to the pulse vaccination rate used here, may perform better. However, setting up this future model would be dependent on having data available about ongoing vaccination rates.

A vaccination rate of 0.005% differs astronomically from the proposed vaccination rate or 41.5% for this flu season. There are a few potential reasons for this phenomenon outside of the scope of this project. The CDC states on their website that the data they have is incomplete. The data is collected from public health laboratories and clinics, requiring individuals to seek medical care in order to be considered in this data set.

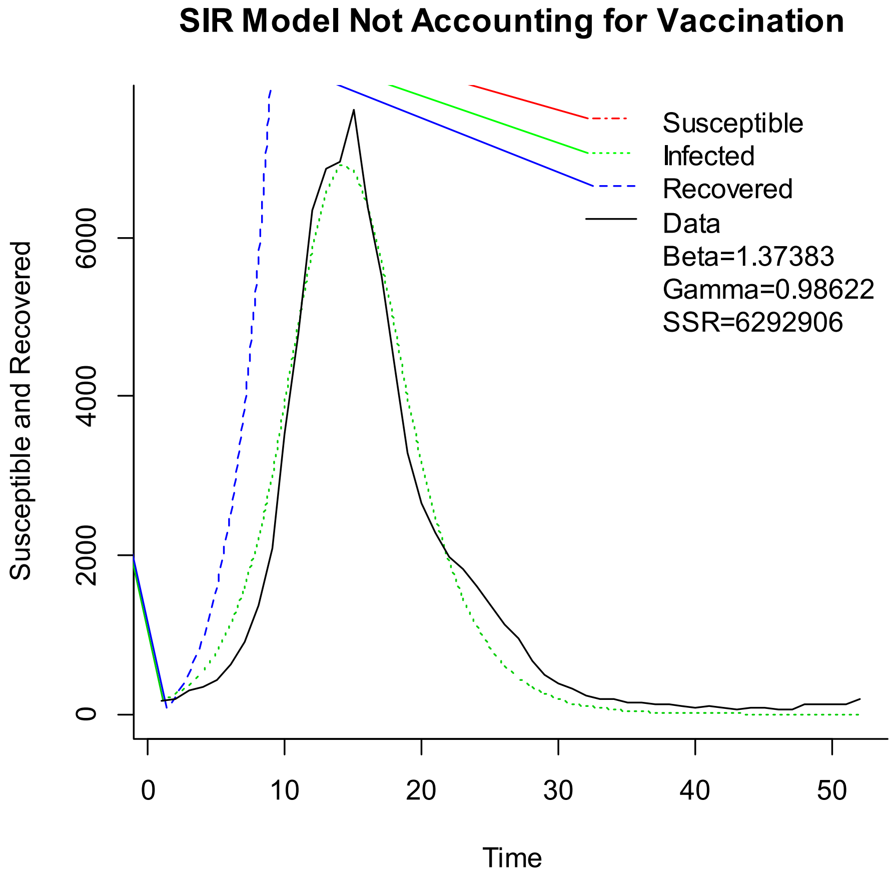

Figure 5. This is the graph of the SIR model fitted to the 2012-2013 flu data without accounting for vaccination rate. The SSR calculated for this model is found in the legend.

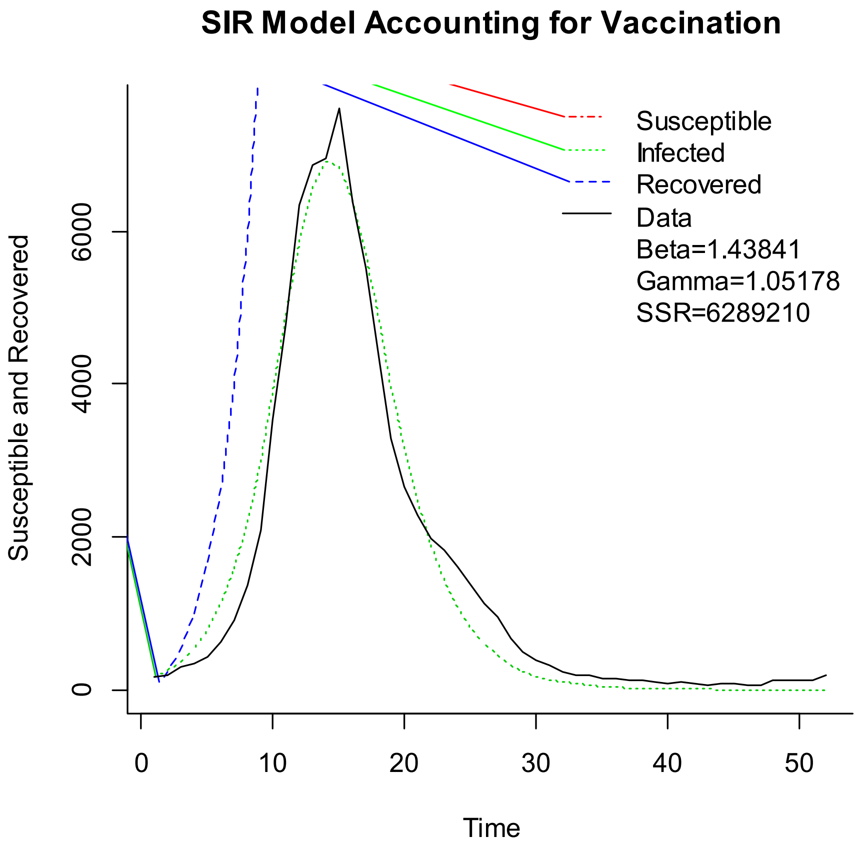

Figure 6. This is the graph of the SIR model fitted to the 2012-2013 flu data that accounts for a vaccination rate of 0.005%. The SSR calculated is in the legend.

Only including data where individuals had to choose to visit public labs or clinics for treatment introduces some problems to consider. People sometimes avoid seeking medical care even when necessary, for reasons such as dissatisfaction with previous medical care, the cost of care, time constraints, or because they suspect they will improve over time regardless.[13] Therefore, the data does not necessarily accurately express the I group in the SIR model. As symptoms of the flu are typical of a minor illness, such as coughing, fatigue, and fever,[14] people likely do not seek medical care. They will most likely get better without seeking explicit medical care, and therefore do not pursue it for the reasons previously listed.

However, aversion to seeking medical care could also mean that the data of vaccination rate also fails to be accurately documented. There is a particular aversion to vaccines, especially with parents being concerned with vaccinating their children. In a poll, around 77% of parents reported concern over at least one childhood vaccine.[15] These reasons are mainly due to religious concerns, personal beliefs (that building a natural immunity is superior to receiving a vaccine), safety concerns, and desire for more information on the vaccine before getting it.[15] Parental concern over vaccinations may influence the vaccination rate, making the true vaccination rate smaller than the CDC's estimation.

Another point of interest is that the CDC's vaccination rate was based on self-report and that they are aware they overestimate the vaccination rate.[16] The CDC website indicates that they know their estimate is generous because the number of doses manufactured for the vaccine is smaller than the amount of people who claim to be vaccinated. In an attempt to further explore this, the number of doses manufactured can be compared to the US population size around the 2012-2013 flu season. For the 2012-2013 flu season, 134.9 million flu vaccine doses were manufactured in the US.[17] Vaccination doses were not considered past March 1st, 2013. The estimated total US population size on March 1st, 2013 was 315.4 million.[18] Therefore, if every single dose made was administer to a person, this would give a vaccination rate of about 43% vaccination rate. This assumes all vaccines were used and none of the vaccines had any extraneous factor that caused their failure, such as delivery or storage problems. This is very close to the CDC's estimated 41.5% vaccination rate for adults. However, their estimate for children is 56.6% were vaccinated in the 2012-2013 season.[16] The US

Census Bureau does attempt to count children as well in the Decennial census and therefore total population size. In 2013, 23.3% of the US population was under the age of 18.[19] Given this information, it can be deduced that there were around 241.9 million adults and 73.5 million children in the US. Using the CDC vaccination rate estimations, 100.4 million adults and 41.6 million children were vaccinated that flu season. This suggests that the vaccination rate for this season cannot be 56.6% for children, since only considering adult vaccination at 41.5% vaccination rate would allocate 100.4 million of the available doses. This would leave 34.5 million of the total 134.9 million doses available for children, making the highest possible vaccination rate 47% for children. Also, using the CDCs vaccination rate estimate and the population size for adults and children, 142 million people got vaccinated. This would be a total vaccination rate of 45% for the US population. Again, there were only 134.9 million doses available, not 142 million. That means these numbers must be an overestimation by at least 7 million. The information on both the adult and child vaccination rate and the number of doses manufactured came from the same source. Between this information and the nature of self-report studies, this indicates a large range in which the true vaccination rate might lie. This suggests that the true vaccination rate may be different from the estimates for it. In addition, there are still other factors to consider when determining vaccination rate.

Other sources that estimate the vaccination rate per year still estimate a much higher percentage than estimated by the SIR model in this paper. However, estimated vaccination rates become significantly lower when focusing research on specific groups of people. It is unclear whether this is due to behaviors common to these groups, or whether closer scrutiny reveals much lower vaccination rates in general. One group that has been studied in relation to flu vaccination rates is the homeless. This is a high-risk group for susceptibility to illness and aversion to seeking medical care. It is estimated that the flu vaccination rate is less than 25% for the homeless in New York City.[20] Given the struggles commonly experienced by the homeless, a low vaccination rate is logical. However, even groups who have easy access to health care are found to have low flu vaccination rate when studied in detail. The CDC estimates that only 36% of healthcare workers chose to receive the flu vaccine. In a case study from the 1990s, only 10% of healthcare workers employed by a long-term care facility reported getting vaccinated for the flu.[21]

To tie vaccination rate to a previous point of interest in this chapter, the residuals between the model accounting for vaccination (with a 0.005% vaccination rate) and the model not accounting for vaccination are very similar. While the model accounting for vaccination has the smaller residuals (and is therefore the better fit), the similar residuals indicate that there is not a huge difference in how well the two models predict the behavior of the data set. This was originally the expected result and indicates that vaccination rate is implicitly accounted for in the original data set.

In summary, previous research supports the idea that the true flu vaccination rate per year is generally overestimated. Given this information and the fact that the very small vaccination rate of 0.005% predicts the behavior of the data, it is not so far stretched to believe the true vaccination rate is much closer to 0.005% than the highest estimate of 56.6%. This is not to say 0.005% is the true vaccination rate, as that is very unlikely. However, these results do emphasize the need for further research. Further indepth research would need to be done where the infectious status of the population is better recorded. Given a more complete data set, the method of accounting for vaccinations using the SIR model described here could be used to estimate the true vaccination rate. This could then be used to both better model influenza data and to better direct preventative measures to decrease the spread and severity of influenza.

VII. DISCUSSION

This paper aimed to apply the SIR epidemiological model to influenza data and defend its use in this context. It also serves to illustrate an area of research that needs to be further expanded upon. These findings defend the use of the SIR model for flu data. They also highlight the uncertainty of the flu vaccination rate for the US within current research. The ultimate goal of this paper was to provide thorough instructions for fitting the SIR model to flu data and to identify potential problems that may arise.

Use of the SIR model is defended in multiple methods throughout this paper. The influenza fits in with the assumptions of the models. Images and figures used show at a glance that the model can closely predict the behavior of the data. This is supported in a more mathematically rigorous manner as well through minimization of the sum of squared residuals (SSR). Using the parameters that produce a model with the smallest SSR ensure the model as accurate as possible. The use of the SIR model in this context is also defended by determining that the model is accurate. This is done by fixing parameter values, like N, and making sure the results of the model do not change. Therefore, the use of the SIR model in this context is shown to be sound.

Fitting the SIR model to flu data highlighted the trouble with accounting for vaccination rate. Estimates for flu vaccination rate in the US during the 2012-2013 flu season were used to show these problems. Firstly, most sources appeared to greatly overestimate the vaccination rate. This leads to the second problem, which is the lack of complete data. The overestimation of the vaccination rate is a product of multiple variables. There is the unreliability of self-report, such as with the CDC’s estimates.

There are also social, health, and educational influences that push parents to not vaccinate their children. The findings in this study for a very low vaccination rate are also found in previous case studies.[20, 21] It is also supported by self-acknowledged limitations within studies estimating the vaccination rate.

For these reasons, the vaccination rate of 0.005% found in this paper is likely to more accurately model reality. However, this vaccination rate is unlikely to be the true vaccination rate in 2012-2013. This would admittedly be a very small vaccination rate and would not make sense in conjunction with self-report studies previous done that suggest a larger rate. This paper does not propose that the true vaccination rate is 0.005% in 2012-2013. The SIR model is a deterministic model, which means it does not have inherent randomness. The world naturally has some inherent randomness. It is also much more complex than this model accounts for. It is also important to note that for these reasons, the best estimate for vaccination rate may not be the true vaccination rate. While 0.005% is very unlikely to be the true vaccination rate, it is the best estimate in this case. This also serves to bring up many questions. As such, this paper does not claim to know the true vaccination rate, but instead serves to show that more research needs to be done on what the true flu vaccination rate per year actually is in the US.

More research on vaccination rate would ideally be done through more modeling using more thorough and complete data. However, this type of data can be very hard to obtain. Therefore, in addition to collecting more complete data, this paper suggests using Bayesian statistics to better model the data and estimate the true parameter values.

VIII: FUTURE RESEARCH SUGGESTION: BAYESIAN STATISTICS

Bayesian Statistics is a data analysis method that could be applied to the influenza data sets. This method is based off Bayes' theorem, which is an equation to calculate conditional probability (how likely a certain event is to occur, while accounting for conditions that may be related to the event) and given in **Equation 3**. When the prior conditions are indeed related to the event of interest, this theorem allows for calculating a more accurate probability that the event will occur. Bayesian statistics is an extension of Bayes' theorem, where prior conditions and information are accounted for when analyzing data.

Equation 3.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Bayesian inference has a general format that it follows. There needs to be a prior distribution, a likelihood function, and a posterior distribution. The prior probability distribution is chosen by the statistician and is the expected distribution of the data. If nothing is known about the data and a distribution cannot be reliably chosen, Jeffreys prior is used (which is a non-informative prior, in order to not affect the results). The likelihood function is based off the data collected, which is conditional to its parameters. The posterior distribution is the distribution of the parameters after taking the observed data into consideration. This differs considerably to the more commonly used Frequentist methods of data analysis.

Bayesian statistics and Frequentist statistics contrast in multiple ways. Frequentist statistics never account for the probability of the hypothesis. The probability of rejecting the null hypothesis compared to the probability of failing to reject the null hypothesis is not considered. Bayesian statistics is also able to better accommodate for latent variables. These are variables that are “hidden” in the data set, and not explicitly observed. For example, vaccination rate is a hidden variable in the flu data sets used. Bayesian is also more resilient to outliers and abnormalities in the data. Another point of interest is that Bayesian models can account for incomplete and missing data.

There are different types of missing data. Some is missed randomly, meaning that there is no relation between the missing data and the observed values. However, some data is missing not at random (MNAR). This is a type of missing data where there is a relationship between the value observed and its likelihood to be missed. For example, people who only experience mild flu symptoms are less likely to go to the doctor and will therefore be less likely to be counted into the data set. Bayesian statistics can accommodate for this by using two different prior probability distributions depending on the likelihood of a participant to be counted in the data set.[22] Next, using Bayesian statistics method, data is simulated. The simulated data can then be used to make better inferences and predictions. Simulated data allows for this because it provides a range of possible data behavior. The data gathered from one season is just one possible way it could potentially behave. This is in part due to the natural variation and stochastic nature of the world. Using Bayesian statistics and simulated data could provide a range of parameter values and their credible intervals, which would better account for inherent randomness.

There has been some previous research done on applying Bayesian statistics to the SIR model. One study analyzed the prevalence of Foot and Mouth disease in cow and sheep farms in England.[23, 24] However, this study was more focused on the infection period for the disease, instead of the infection rate. As a result, their data simulation was based around analyzing the variance in time, rather than around analyzing the amount of farm animals infected. Another study looked at respiratory syncytial virus in Spain using the SIRS model.[25] This model is comparable to the SIR model, but people who have recovered from the disease will eventually become susceptible to it again. Their simulated data used a discrete time measurement. They focused their simulation to further analyze the amount of weekly hospital visits, which is a discrete time model. This is similar to how the influenza data sets would need to be simulated. A binomial distribution was used as the prior probability distribution, with the number of infections at a given time and the likelihood of hospitalization as the parameter values. The number of infections was also modeled with a binomial distribution, with the amount of susceptible people and the infection rate as the parameters. This information was then plugged into the SIRS model and the authors used Markov chain Monte Carlo simulation techniques in order to simulate the data. This allows for more accurately predicting the timing and magnitude of respiratory syncytial virus epidemics, as well as better inference by quantifying the uncertainty surround the SIRS parameters.

The use of Bayesian statistics to model the respiratory syncytial virus is a similar application to the SIR model application suggested by this paper. Right now, there is not very much research done on applying Bayesian statistics to the SIR model, particularly not in conjunction to analyzing influenza data from the US. This would be an important next step because a better understanding of influenza data can allow for more effective preventative measures to be taken. This would in turn result in better general health and reduce the socio-economic effects the influenza has on the US each year.

REFERENCES

1. Durand, L. O., Cheng, P., Palekar, R., Clara, W., Jara, J., Cerpa, M., Balmaseda, A. (2016). Timing of influenza epidemics and vaccines in the American tropics, 2002-2008, 2011-2014. Influenza & Other Respiratory Viruses, 10(3), 170–175. https://doi-org.prxy4.ursus.maine.edu/10.1111/irv.12371

2. Reed, C., Chaves, S. S., Kirley, P. D., Emerson, R., Aragon, D., Hancock, E. B., Finelli, L. (2015, March 4). Estimating Influenza Disease Burden from Population-Based Surveillance Data in the United States. Retrieved from https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0118369

3. Mao, L., Yang, Y., Qiu, Y., & Yang, Y. (2012, May 17). Annual economic impacts of seasonal influenza on US counties: Spatial heterogeneity and patterns. Retrieved from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3479051/#B3

4. Smith, D., & Moore, L. (2004). The SIR Model for Spread of Disease - The Differential Equation Model. Retrieved from https://www.maa.org/press/periodicals/loci/joma/the-sir-model-for-spread-of-disease-the-differential-equation-model

5. Jones, J. H. (2007, May 1). Notes on R0. Retrieved from https://web.stanford.edu/~jhj1/teachingdocs/Jones-on-R0.pdf

6. Kock, K. D., Tavares, E. G., Traebert, J. L., & Maurici, R. (2017, January 17). Calculation of reproducibility rates (R0) by simplification of SIR model applied to Influenza A epidemic (H1N1) in Brazil occurred in 2009. Retrieved from https://doaj.org/article/d84cb7c0be044a189535722834b3a4d9

7. O'Regan, S.M., Kelly, T.C., Korobeinikov, A. et al. J. Math. Biol. (2013) 67: 293. https://doi-org.prxy4.ursus.maine.edu/10.1007/s00285-012-0550-9

8. Sambaturu, N., Mukherjee, S., López-García, M., Molina-París, C., Menon, G. I., & Chandra, N. (2018, March 18). Role of genetic heterogeneity in determining the epidemiological severity of H1N1 influenza. Retrieved from https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1006069

9. National, Regional, and State Level Outpatient Illness and Viral Surveillance. (n.d.). Retrieved from https://gis.cdc.gov/grasp/fluview/fluportaldashboard.html

10. Hhs.gov. (2018, November 14). Attachment B: Considerations and Recommendations concerning. Retrieved from https://www.hhs.gov/ohrp/sachrp-committee/recommendations/2013-may-20-letter-attachment-b/index.html

11. University of Maine. (2019, January 25). Policies and Procedures for the Protection of Human Subjects of Research. Retrieved from https://umaine.edu/research-compliance/human-subjects/guidance-policy/

12. Cdc.gov. (2017, November 01). Influenza (Flu). Retrieved from https://www.cdc.gov/flu/fluvaxview/coverage-1516estimates.htm

13. Taber, J. M., Leyva, B., & Persoskie, A. (2014). Why do people avoid medical care? A qualitative study using national data. *Journal of general internal medicine*, *30*(3), 290–297. doi:10.1007/s11606-014-3089-1

14. Cdc.gov. (2019, March 15). Influenza (Flu). Retrieved from https://www.cdc.gov/flu/consumer/symptoms.htm

15. McKee, C., & Bohannon, K. (2016). Exploring the Reasons Behind Parental Refusal of Vaccines. *The journal of pediatric pharmacology and therapeutics: JPPT: the official journal of PPAG*, *21*(2), 104–109. doi:10.5863/1551-6776-21.2.104

16. Cdc.gov. (2016, June 23). Influenza (Flu). Retrieved from https://www.cdc.gov/flu/fluvaxview/coverage-1415estimates.htm#data

17. Cdc.gov. (2018, August 23). Influenza (Flu). Retrieved from https://www.cdc.gov/flu/professionals/vaccination/vaccinesupply.htm

18. Census.gov. (n.d.). U.S. and World Population Clock. Retrieved from https://www.census.gov/popclock/

19. Hrsa.gov. (n.d.). Population of Children. Retrieved from https://mchb.hrsa.gov/chusa14/population-characteristics/population-children.html

20. Buchner, S., Brickner, P., & Vincent, R.(2006). Influenza illness among homeless persons. *Emerging Infectious Diseases*, 12(7), 1162–1163.

21. Nelson, R. (2004). AJN Reports: Health Care Workers: Few Feel the Flu Shot. *The American Journal of Nursing, 104*(10), 24-25. Retrieved from http://www.jstor.org.prxy4.ursus.maine.edu/stable/29746157

22. Kopra, J., Karvanen, J., & Tommi Harkanen, T. (2017, June 20). Bayesian models for data missing not at random in health examination surveys. Retrieved from https://arxiv.org/pdf/1610.03687.pdf

23. Peck, R. (2015). Efficient and Adaptive MCMC and Simulation Based Methods for Spatial Epidemics.

24. Peck, R. (2015). *Bayesian Inference for a SIR Epidemic Model*.

25. Corberán-Vallet, A., & Santonja, F. J. (2014, August 4). A Bayesian SIRS model for the analysis of respiratory syncytial virus in the region of Valencia, Spain. Retrieved from https://onlinelibrary-wiley-com.prxy4.ursus.maine.edu/doi/full/10.1002/bimj.201300194

APPENDIX

sir.model():

```
setwd("C:/Users/…")
library(deSolve)


data<-read.csv("2017-18 public health lab.csv")
data$Notes<-NULL
total.people <- c(data$TOTAL.SPECIMENS[1:52])
total.i <- c(data$Total.Infected[1:52])
total.s <- total.people-total.i

#start of I and R at start
init.v <- c(I=total.i[1],R=0)
state.value <- c(I=total.i[1],R=0)

# Time frame
t <- length(data$WEEK)
data$Time <- 1:t
time.v <- data$Time[1:52]


##Start of function

sir.model <- function(parms.v=c(1, 0.1, total.s[1]), init=init.v, times=time.v,
label.t="SIR Model 2017-2018"){

#SIR function EQUATIONS
sir.equation <- function(times=time.v, state.value=state.value, parms=parms.v) {
```

```
β <- parms[1]
γ <- parms[2]
S <- state.value[1]
I <- state.value[2]
R <- state.value[3]

  with(as.list(c(state.value, parms)), {

#β is transmission rate
#γ is recovery rate


        N <- S+I+R
        dS <- -β*((S*I)/N)
        dI <- (β*((S*I)/N))-(γ*I)
        dR <- γ*I

        return(list(c(dS, dI, dR)))
  })
}


## Solve using ode (General Solver for Ordinary Differential Equations)
out <- ode(y = c(parms.v[3],init.v), times = time.v, func = sir.equation,
parms=c(parms.v[1],parms.v[2]))

out <- as.data.frame(out)
# Delete time variable
out$time <- NULL
# Show data
head(out, 10)
```

```
out$N <- NULL

#Plot
i.v <- c(data$Total.Infected[1:52]) #actual data

matplot(x = time.v, y = out, type = "l", xlab = "Time by Weeks", ylab = "Number of
Individuals", main = label.t,lwd = 1, lty = c(4,3,2), bty = "l", col = c(2:4, "black"),
ylim=c(0,max(i.v)), xlim=c(0,52))
legend("topright", c("Susceptible", "Infected", "Recovered", "Data", "Beta =
1.85896","Gamma = 1.55068","Initial S = 287181.6"), col = c(2:4, "black"), bty = "n",
lty = c(4,3,2,1,0,0,0))
lines(y=i.v, x=time.v, col="black")
return(out)
}



## The below line of code uses the SIR equations to produce results,
## which are estimates of how the data acts over time
# Solve using ode (General Solver for Ordinary Differential Equations)
out <- ode(y = init, times = time.v, func = sir.equation, parms=parms.v)

out <- as.data.frame(out)
# Delete time variable
out$time <- NULL
# Show data
head(out, 10)

#Plot
i.v <- total.i  #actual data
```

```
matplot(x = time.v, y = out, type = "l", xlab = "Time", ylab = "Susceptible and
Recovered", main = "SIR Model",lwd = 1, lty = 1, bty = "l", col = 2:4, ylim=c(0,3500))
legend(40, 0.7, c("Susceptible", "Infected", "Recovered"), pch = 1, col = 2:4, bty = "n")
lines(y=i.v, x=time.v)

return(out)
}
sir.model()
```

L1():

```
##This needs the same setup before the function as the sir.model(), however that is
omitted ##here to avoid excessive repetition

#Call our sir.model() function in order to use it in our L1() function
source("C:\\...\\sir model.R")

L1 <- function(parms.v=c(0.3,0.1,total.people[1]), times=time.v, I.v=i.v, init=init.v) {

        β <- parms.v[1]
        γ <- parms.v[2]
        N <- parms.v[3]

        #Retrieving the estimate I values over time
        out <- sir.model(parms.v, init, time.v)
        odeI.v <- out$I

        #Calculating the SSR
        res.v <- I.v - odeI.v
        return(sum(res.v^2))
```

```
    }

    #Finding the best parameter values
    par.v=optim(c(.3,0.1,1139),fn=L1, times=time.v, I.v=i.v, init=init.v)$par
    print(par.v)

    #Best B estimate
    p1 <- par.v[1]
    #Best Γ estimate
    p2 <- par.v[2]
    #Best N estimate
    p3 <- par.v[3]
```

Heat Map:

```
#call our L1() function in order to use it within this code
source("C:\\...\\L1 optimizing.R")

resolution = 100
β.v = seq(0, 4, len=resolution)
γ.v = seq(0, 4, len=resolution)
# rows are β, cols are γ
A.m = matrix(NA, nrow=resolution, ncol=resolution)
for (βIndex in 1:resolution) {
  β = β.v[βIndex]
  for (γIndex in 1:resolution) {
    γ = γ.v[γIndex]
    A.m[βIndex,γIndex] = L1(c(β,γ,S=2.871816e+05), time.v, i.v, init.v)
  }
}
x <- seq(min(β.v),max(β.v), length=nrow(A.m))
```

```
y <- seq(min(γ.v),max(γ.v),length=ncol(A.m))
image(log(A.m),col=rainbow(500), x=x, y=y, xlim=c(0,4), ylim=c(0,4))
grid(col="black")
points(1.85896, 1.55068, pch=8)
points(2, 1.7)
points(2.4, 2.05)
points(2.125, 1.8)
points(1.6,1.33)
points(1.75, 1.45)

library(lattice)
xlim <- list(at=seq(min(β.v),max(β.v),by=0.5))
ylim <- list(at=seq(min(γ.v),max(γ.v),by=0.5))
levelplot(log(A.m), row.values=β.v, column.values=γ.v, aspect="fill", xlab="β",
ylab="γ", scales=list(x=xlim, y=ylim))
```

sir.model.vacc

```
##Chapter III method
library(deSolve)

data<-read.csv("2012-13.csv")

total.people <- c(data$TOTAL.SPECIMENS)
total.i <- c(data$Total.Infected)
total.s <- total.people-total.i
i.v <- c(data$Total.Infected)
# Time frame
t <- length(data$WEEK)
```

```
data$Time <- 1:t
time.v <- data$Time

vacc.percent <- 0.005

#start of I and R at start
init.vU1 <- c(I=total.i[1],R=0)
state.valueU1 <- c(I=total.i[1],R=0)

###SIR FUNCTION TO FIND S VALUE UNVACC
sir.model.U1 <- function(parms.v=c(1, 0.1, total.s[1]), init=init.vU1, times=time.v){

#SIR function EQUATIONS
sir.equation <- function(times=time.v, state.value=state.valueU1, parms=parms.v) {

β <- parms[1]
γ <- parms[2]
S <- state.value[1]
I <- state.value[2]
R <- state.value[3]

  with(as.list(c(state.value, parms)), {

#β is transmission rate
#γ is recovery rate

            N <- S+I+R
            dS <- -β*((S*I)/N)
            dI <- (β*((S*I)/N))-(γ*I)
            dR <- γ*I
```

```
        return(list(c(dS, dI, dR)))
  })
}


## Solve using ode (General Solver for Ordinary Differential Equations)
out <- ode(y = c(parms.v[3],init.vU1), times = time.v, func = sir.equation,
parms=c(parms.v[1],parms.v[2]))

out <- as.data.frame(out)
out$time <- NULL
head(out, 10)
out$N <- NULL

#Plot
i.v <- c(data$Total.Infected) #actual data
matplot(x = time.v, y = out, type = "l", xlab = "Time", ylab = "Susceptible and
Recovered", main = "SIR Model",lwd = 1, lty = 1, bty = "l", col = 2:4,
ylim=c(0,max(data$Total.Infected)))
legend("topright", c("Susceptible", "Infected", "Recovered", parms.v), pch = 1, col =
2:4, bty = "n")
lines(y=i.v, x=time.v)

return(out)
}

###L1 FUNCTION 1
L1.U1 <- function(parms.v=c(1, 0.1, total.s[1]), times=time.v, I.v=i.v, init=init.vU1) {

β <- parms.v[1]
γ <- parms.v[2]
```

```
S <- parms.v[3]

out <- sir.model.U1(parms.v, init, time.v)
odeI.v <- out$I
res.v <- I.v - odeI.v
return(sum(res.v^2))
}
par.vU1=optim(c(1, 0.1, total.s[1]),fn=L1.U1, times=time.v, I.v=i.v, init=init.vU1,
upper=c(20,20,1000000), lower=c(0,0,0), method="L-BFGS-B")$par
#print("Parms for Unvacc SIR model to Find Init S")
#print(par.vU1)
R2.U1 <- L1.U1(parms.v=c(1, 0.1, total.s[1]), times=time.v, I.v=i.v, init=init.vU1)
#print("Sum of R^2")
#print(R2.U1)

##Chapter V method
### SIR MODEL UNVACC WITH UPDATED S

#start of S I and R at start
#S is average of optim S values
init.vU2 <- c(S=par.vU1[3],I=total.i[1],R=0)
state.valueU2 <- c(S=par.vU1[3],I=total.i[1],R=0)

sir.model.U2 <- function(parms.v=c(par.vU1[1], par.vU1[2]), init=init.vU2,
times=time.v){

#SIR function EQUATIONS
sir.equation <- function(times=time.v, state.value=state.valueU2, parms=parms.v) {

β <- parms[1]
γ <- parms[2]
S <- state.value[1]
I <- state.value[2]
R <- state.value[3]

  with(as.list(c(state.value, parms)), {

#β is transmission rate
#γ is recovery rate


          N <- S+I+R
          dS <- -β*((S*I)/N)
          dI <- (β*((S*I)/N))-(γ*I)
          dR <- γ*I

          return(list(c(dS, dI, dR)))
  })
}

## Solve using ode (General Solver for Ordinary Differential Equations)
out <- ode(y = init.vU2, times = time.v, func = sir.equation, parms=parms.v)

out <- as.data.frame(out)
out$time <- NULL
head(out, 10)
out$N <- NULL

#Plot
i.v <- c(data$Total.Infected) #actual data
matplot(x = time.v, y = out, type = "l", xlab = "Time", ylab = "Susceptible and
Recovered", main = "SIR Model Not Accounting for Vacciantion",lwd = 1, lty = 1, bty
= "l", col = 2:4, ylim=c(0,max(data$Total.Infected)))
```

```
legend("topright", c("Susceptible", "Infected", "Recovered", parms.v), pch = 1, col =
2:4, bty = "n")
lines(y=i.v, x=time.v)

return(out)
}

###L1 2
L1.U2 <- function(parms.v=c(par.vU1[1], par.vU1[2]), times=time.v, I.v=i.v,
init=init.vU2) {

β <- parms.v[1]
γ <- parms.v[2]
out <- sir.model.U2(parms.v, init, time.v)
odeI.v <- out$I
res.v <- I.v - odeI.v
return(sum(res.v^2))
}

par.vU2=optim(c(par.vU1[1], par.vU1[2]),fn=L1.U2, times=time.v, I.v=i.v,
init=init.vU2, upper=c(20,20,1000000), lower=c(0,0,0), method="L-BFGS-B")$par
#print("Parms for Unvacc SIR Model Updated Init S")
#print(par.vU2)
R2.U2 <- L1.U2(parms.v=c(par.vU1[1],par.vU1[2]), times=time.v, I.v=i.v,
init=init.vU2)
#print("Sum of R^2")
#print(R2.U2)


##Chapter III method
### SIR TO FIND S VALUE VACC
```

```
#start of I and R at start
init.vV1 <- c(I=total.i[1],R=(vacc.percent/(100-vacc.percent))*total.s[1])
state.valueV1 <- c(I=total.i[1],R=(vacc.percent/(100-vacc.percent))*total.s[1])

###SIR MODEL VACC TO FIND S
sir.model.V1 <- function(parms.v=c(1, 0.1, total.s[1]), init=init.vV1, times=time.v,
label.t="SIR Model 2012-13"){

#SIR function EQUATIONS
sir.equation <- function(times=time.v, state.value=state.valueV1, parms=parms.v) {

β <- parms[1]
γ <- parms[2]
S <- state.value[1]
I <- state.value[2]
R <- state.value[3]

  with(as.list(c(state.value, parms)), {

#β is transmission rate
#γ is recovery rate


          N <- S+I+R
          dS <- -β*((S*I)/N)
          dI <- (β*((S*I)/N))-(γ*I)
          dR <- γ*I

          return(list(c(dS, dI, dR)))
  })
}

## Solve using ode (General Solver for Ordinary Differential Equations)
out <- ode(y = c(parms.v[3],init.vV1), times = time.v, func = sir.equation,
parms=c(parms.v[1],parms.v[2]))

out <- as.data.frame(out)
out$time <- NULL
head(out, 10)
out$N <- NULL

#Plot
i.v <- c(data$Total.Infected) #actual data
matplot(x = time.v, y = out, type = "l", xlab = "Time", ylab = "Susceptible and
Recovered", main = label.t,lwd = 1, lty = 1, bty = "l", col = 2:4,
ylim=c(0,max(data$Total.Infected)), xlim=c(1,52))
legend("topright", c("Susceptible", "Infected", "Recovered", parms.v), pch = 1, col =
2:4, bty = "n")
lines(y=i.v, x=time.v)

return(out)
}

L1.V1 <- function(parms.v=c(1, 0.1, total.s[1]), times=time.v, I.v=i.v, init=init.vV1) {
β <- parms.v[1]
γ <- parms.v[2]
S <- parms.v[3]
out <- sir.model.V1(parms.v, init, time.v)
odeI.v <- out$I
res.v <- I.v - odeI.v
return(sum(res.v^2))
```

```
}
par.vV1=optim(c(1, 0.1, total.s[1]),fn=L1.V1, times=time.v, I.v=i.v, init=init.vV1,
upper=c(20,20,1000000), lower=c(0,0,0), method="L-BFGS-B")$par
#print("Parms for Vacc SIR Model to Find Init S")
#print(par.vV1)
R2.V1 <- L1.V1(parms.v=c(1, 0.1, total.s[1]), times=time.v, I.v=i.v, init=init.vV1)
#print("Sum of R^2")
#print(R2.V1)


##Chapter V method
###SIR MODEL VACC WITH UPDATED S

#start of S I and R at start
#S is average of optim S values
#adjust s by 41.5% b/c thats the vaccination estimate per adults
init.vV2 <- c(S=(par.vV1[3]*(1-
(0.01*vacc.percent))),I=total.i[1],R=(par.vV1[3]*(0.01*vacc.percent)))
state.valueV2 <- c(S=par.vV1[3]*(1-
(0.01*vacc.percent)),I=total.i[1],R=(par.vV1[3]*(0.01*vacc.percent)))
#https://www.cdc.gov/flu/fluvaxview/coverage-1516estimates.htm

###SIR VACC UPDATED S
sir.model.V2 <- function(parms.v=c(par.vV1[1], par.vV1[2]), init=init.vV2,
times=time.v){

#SIR function EQUATIONS
sir.equation <- function(times=time.v, state.value=state.valueV2, parms=parms.v) {

β <- parms[1]
γ <- parms[2]
N <- state.value[1]
I <- state.value[2]
R <- state.value[3]
S <- (1-(0.01*vacc.percent))*N
  with(as.list(c(state.value, parms)), {

#β is transmission rate
#γ is recovery rate


            N <- S+I+R
            dS <- -β*((S*I)/N)
            dI <- (β*((S*I)/N))-(γ*I)
            dR <- (γ*I)

            return(list(c(dS, dI, dR)))
  })
}

## Solve using ode (General Solver for Ordinary Differential Equations)
out <- ode(y = init.vV2, times = time.v, func = sir.equation, parms=parms.v)

out <- as.data.frame(out)
out$time <- NULL
head(out, 10)
out$N <- NULL

#Plot
i.v <- c(data$Total.Infected) #actual data
matplot(x = time.v, y = out, type = "l", xlab = "Time", ylab = "Susceptible and
Recovered", main = "SIR Model Accounting for Vaccination",lwd = 1, lty = 1, bty =
"l", col = 2:4, ylim=c(0,max(data$Total.Infected)))
```

```
legend("topright", c("Susceptible", "Infected", "Recovered", parms.v), pch = 1, col =
2:4, bty = "n")
lines(y=i.v, x=time.v)

return(out)
}

L1.V2 <- function(parms.v=c(par.vV1[1], par.vV1[2]), times=time.v, I.v=i.v,
init=init.vV2) {

β <- parms.v[1]
γ <- parms.v[2]
out <- sir.model.V2(parms.v, init, time.v)
odeI.v <- out$I
res.v <- I.v - odeI.v
return(sum(res.v^2))
}

par.vV2=optim(c(par.vV1[1], par.vV1[2]),fn=L1.V2, times=time.v, I.v=i.v,
init=init.vV2, upper=c(20,20), lower=c(0,0), method="L-BFGS-B")$par
#print("Parms for Vacc SIR Model Updated Init S")
#print(par.vV2)
R2.V2 <- L1.V2(parms.v=c(par.vV1[1], par.vV1[2]), times=time.v, I.v=i.v,
init=init.vV2)
#print("Sum of R^2")
#print(R2.V2)


#########################
##The section of code below will give the R^2 results
## in a quick and easy to read format
```

```
if (R2.U2 < R2.V2) {

print("UnVacc model has a smaller sum(R^2)")
print("UnVacc")
print(R2.U2)
print("Vacc")
print(R2.V2)

} else if (R2.U2 > R2.V2) {

print("Vacc model has smaller sum(R^2)")
print("UnVacc sum(R^2)")
print(R2.U2)
print("Vacc sum(R^2)")
print(R2.V2)

}

print(R2.V2-R2.U2)
```

AUTHOR'S BIOGRAPHY

Madeline Isabelle Dorr was born in Virginia on October 28, 1997. She was raised in the Northern Virginia area and stayed there until she graduated high school in 2015. After high school, she went to Maine to attend the University of Maine at Orono. Taking advantage of the snowfall in Maine, she spent a lot of Saturdays during her freshman and sophomore year at Sugar Loaf skiing. Currently, she stays active by walking her adorable dog, Sadie.

Majoring in mathematics and statistics, she also has minors in psychology and neuroscience. Upon graduation, she is moving back to the Northern Virginia area to get her master's in biostatistics at George Washington University in DC.