Lab/Homework #1

HUMBIO 154D/HRP 204

Due Date: April 23, 2020

**Instructions:** This homework assignment is designed to accompany the first in-class lab on 4/21/20 and to be partially completed during lab. You should be able to complete all of exercises using equations defined in lecture and from assigned reading in Keeling & Rohani, and model and plotting functions available on Canvas (see Lab1\_functions.R) and demoed in lab 1. While we will grade this homework assignment based on completion, you should use this as an opportunity to concretize what you have learned in class and labs so far. Future homework assignments and labs will build on what you learn while completing this assignment. You are welcome to work with your classmates but we ask everyone to submit their own responses (as a Word doc) and their own accompanying R code to Canvas.

First, complete steps 1-5 below in R. The code has been filled in for you. You do not need to submit any responses to questions 1-5. Include responses (in the form of plots, calculated quantities, and short written responses to questions) for questions 6 on. Note that questions 15 and 16 are optional.

**Initialization Steps (with accompanying code – check your output against the output included for question 5 before starting questions 6 on):**

1. Load packages

#if you are installing packages for the first time:  
#install.packages("deSolve")  
#install.packages("ggplot2")  
  
library(deSolve) #differential equation solver  
library(ggplot2) #plotting package

2. Start by defining a vector of model parameters (e.g. effective contact rate, recovery rate) and a vector of initial compartment sizes (often 1 infected person and the rest susceptible). We also define a vector of time steps corresponding to how long we want to run the model.

parameters <- c(beta = 0.5, #effective contact rate (aka transmission rate)  
 gamma = 0.3 #recovery rate (1/duration infection)  
)  
  
state <- c(S = 99999, #population of 100,000, 1 person starts of infected  
 I = 1,   
 R = 0  
)  
  
T\_end <- 500 #run model for 500 time steps  
times <- seq(0, T\_end, by = 1) #runs the model for 500 time steps, and computes output at each time step

Note: Your time steps could be days, months, years - but make sure time step size matches time scale of parameters. For example, if your time step is days, make sure you are using daily rates.

Note: by naming the components of state (e.g. S=99999), and parameters (e.g. beta=0.5), R can match them with the names of parameters and states in your SIR model function (*BasicSIR)* below.

3. Define SIR (and related) model functions. These will be used with the deSolve package to simulate how your population moves between compartments (e.g. Susceptible, Infected, Recovered) over time, given a set of parameters and the initial compartment sizes that we defined in step 2. We’ll start by writing a function for a basic SIR model without demography. For this and all subsequent models, we will assume frequency dependence (not density dependence).

BasicSIR<-function(t, state, parameters) {  
 with(as.list(c(state, parameters)),{ #this tells R that "S" refers to the "S" in the "state" vector, "beta" refers to the "beta" in "parameters", etc.  
   
 N = S + I + R #define N (total population size)  
   
 #SIR model equations from lecture - rates of change in and out of each compartment   
 dS <- -beta\*S\*I/N  
 dI <- beta\*S\*I/N - gamma\*I  
 dR <- gamma\*I  
   
 #return the rates of change as a list  
 list(c(dS, dI, dR))   
 })  
}

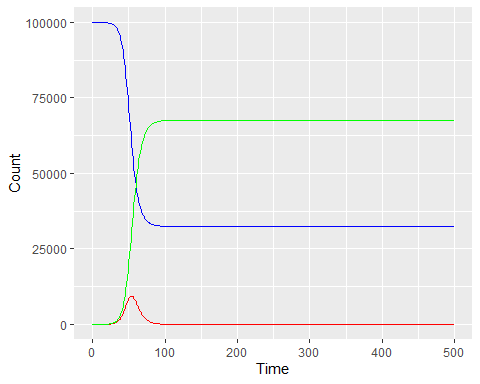
4. We’ve defined our SIR model function and its inputs. Now we can run it using the *ode* function that is part of the deSolve package.

output <- ode(y = state, times = times, func = BasicSIR, parms = parameters)

Note: *ode()* is a function in the deSolve package. You must fill it in using the syntax: “ode(y= [vector of initial compartment sizes], times = [vector of time steps], func = [name of SIR model function], parms = [vector or list of parameter values])”

5. We can examine the model output by typing “View(output)” in the console, clicking on “output” in the Environment (top right of RStudio), or printing model output to the console. It is often easier to examine model output by plotting. We start by defining a function that will do basic cleaning and plotting the epidemic curve from our model output from step #4.

show\_SIR\_model\_results<-function(output) {  
 df1 <- data.frame(output)  
 ggplot() +   
 geom\_line(data = df1, aes(x = time, y = S), color = "blue") +  
 geom\_line(data = df1, aes(x = time, y = I), color = "red") +  
 geom\_line(data = df1, aes(x = time, y = R), color = "green") +  
 xlab('Time') +  
 ylab('Count')   
}  
  
show\_SIR\_model\_results(output)



print(tail(output, 1)) #tail(object, x) refers to the last x rows of an object

## time S I R  
## [501,] 500 32423.62 4.797714e-22 67576.38

print(output[T\_end+1,]) #501 is the last row because the initial compartment sizes are saved in the 1st row (t=1) and then the model runs for 500 time steps

## time S I R   
## 5.000000e+02 3.242362e+04 4.797714e-22 6.757638e+04

**Questions (submit your responses and code for 6-14. 15 and 16 are optional)**

6. What is the basic reproductive number, R0, for this model (use the equations from lecture)? What is the population prevalence at t=50 (obtain this from your model output)? Is there an endemic equilibrium? Why/why not?

7. How does the basic reproductive number change if we change beta to parameterize a contact rate (k) of 10 contacts per unit time and a 20% probability of infection per contact (p)? How do you expect this to change the initial epidemic dynamics and proportion of the population that gets infected? Plot the epidemic curve graph with this new beta and compare against your prediction.

8. If we wanted to model a chronic infectious disease, what type of model would we use instead? Is there a way to model this using the *BasicSIR* function or do we need to write a new model function? Generate the epidemic curve plot for a chronic disease with beta=0.5. Does the population get infected faster or slower than in the previous version (part 5) and do more or fewer people get infected? Conceptually, why is this?

9. Now let’s add demography. To do this, we need to write a new model function (hint – use the *OpenSIR* function from Lab1\_functions.R) that includes births and deaths and we need to define additional parameters. You can set the birth rate and the death rate both to 0.03. Keep beta=0.5 and gamma=0.3 as before.

In addition to births and deaths, we’ve added another new parameter - omega. We’ll set omega to 0 for now, but what does omega represent?

10. Calculate R0 and the compartment sizes at the endemic equilibrium using the equations from lecture. Do we expect the system to reach the endemic equilibrium or the disease-free equilibrium, and why? Did R0 increase or decrease compared to the *BasicSIR* model (without demography - part 6)? What is your intuition behind this?

11. Plot the epidemic curve of this new model and examine the modeled equilibrium compartment sizes. Compare this against your prediction.

12. We can also plot the phase diagram (prevalence of S against prevalence of I) to observe how the system reaches equilibrium (as we showed in lecture 3). We define an additional graphing function for this - *phase\_diagram* (available in the Lab1\_functions.R file).

13. Calculate the approximate average age of infection for this model and the approximate period of oscillation using the equations from lecture. Does the estimated period of oscillation look similar to the oscillation in the graph from question 11? Assume that time steps (and rates) are in months.

14. One at a time (holding all other parameters fixed), make the following changes:

a.) change beta to 2.0 (keep gamma at 0.3, births and deaths at 0.03)

b.) change gamma to 0.2 (keep beta at 0.5, births and deaths at 0.03)

[question continues onto following page]

c.) change births and deaths (both) to 0.1 (keep beta at 0.5 and gamma at 0.3).

Plot the resulting epidemic curves and phase diagrams. How do these changes affect the infection dynamics? Why?

**Optional:** 15. Let’s revisit the omega parameter we added to *OpenSIR.* This parameter represents waning immunity (recall - when omega=0, immunity is lifelong). What type of model do we have as omega approaches infinity?

**Optional:** 16. Set omega to equal 0.1. Run the model and plot the resulting epidemic curve. What do you notice? Has the R0 changed (from part 10)? If not, how do you explain the difference in output? (hint: think closely about the definition of R0)