Homework 1

Released: Aug 29, 2024; Due: **5pm ET, Sept 10, 2024**

Georgia Tech
College of Computing
B. Aditya Prakash

CSE 8803 EPI Fall 2024
NAME: Shikhar Verma
GTID: 903497421

Days Late: 1 day

Slip Time Left: 3 days

Reminders:

1. *Out of 100 points. 5 Questions.*
2. *If you use Late days, mark how many you are using (out of maximum 4 available) at the top of your answer PDF.*
3. *There could be more than one correct answer. We shall accept them all.*
4. *Whenever you are making an assumption, please state it clearly.*
5. *You will submit a solution pdf LASTNAME.pdf containing your answers and the plots as well as a tar-ball LASTNAME.tgz that contains your code and any output files.*
6. *Please type your answers either in LaTeX document or in a separate file like a Word document and then convert it into a pdf file. Typed answers are strongly encouraged. Illegible handwriting may get no points,at the discretion of the grader. Only drawings may be hand-drawn, as long as they are neat and legible.*
7. *Additionally, you will submit one tar-ball LASTNAME.tgz that contains your code and any results files. Code and results for each question should be contained in a separate sub-directory (Eg: Q1) and there should be a README.txt file for each sub-directory explaining any packages to install, command to run the code files and location of the expected output. Please follow the naming convention **strictly**.*
8. *If a question asks you to submit code please enter the file path (Eg: Q1/Q-1.3.1.py) in the solution pdf.*
9. *You can download all the datasets needed for this homework from canvas files, you can check the information about the datasets in the README.txt file.*

1. (50 points) SIR Model

Q 1.1 (5 points) **Everyone loves DS for Epi.** It's the beginning of the Fall 2024 semester. Initially, only 1% of the student population loves doing *Data Science for Epidemiology* (DSE). However, it is very contagious: once someone becomes interested in DSE they try to spread their interest to any uninterested people for a short time before they can no longer do so. Interest in DSE is also very persistent: those who develop interests in DSE stay that way. We model the spread of interest in DSE using a SIR model: "Susceptible" students are uninterested, "Infected" students recently caught the interest and can spread it to others, and "Recovered" students are interested but can't spread their interests anymore.

We observe that at the end of a very long semester ($t \to \infty$) 80% of the student population are interested in DSE. What is the expected fraction of fellow students an interested (infected) student spreads their interest to?

Hint: The problem asks for calculating the reproductive number (R_0) in a SIR model. You may use the results from the slides without deriving the ODE solution.

Hint: You may assume the initial infected 1% are all still contagious. (i.e., $R(0) = 0$)

Solution:
https://drive.google.com/file/d/16Bq21eMfHmpnbeEz6xn_eOL1yt5oIT8q/view?usp=sharing

Q 1.2 (13 points) **ODE ⇔ Networks** Let us try to derive the SIR ODE model from the SIR network model. Assume that the network is a clique of size N (i.e. everyone is connected to everyone else with a total population of N individuals). Further, assume the model evolves in discrete time steps. At each step:

1. each susceptible individual u picks a person v from its neighbors, uniformly randomly
2. if v is infected, then u gets infected with probability β.
3. each infected individual goes into the R state with probability γ

Let $X(t)$ be the number of individuals who are in the S state at a time t. Similarly, $Y(t)$ is the number of individuals who are in state I, and $Z(t)$ is the number of individualsin state R.

Hint: This SIR network model is slightly different from that described in class which was from the point of view of the infected node. Here, we describe from the point of view of a susceptible node but will produce an equivalent result.

Hint: You may assume $N \approx N - 1$ in derivation, as population base is typically large.

Q 1.2.1 (2 points) What is the probability $P(t)$ of a single susceptible individual getting infected at time t? Clearly the total number of expected new infections at time $t+1$ will be $P(t) \times X(t)$.

Solution:

https://drive.google.com/file/d/1mEkQwuHcqPyGKSVLlhXF9r2Z1UiFmpbT/view?usp=sharing

Q 1.2.2 (4 points) What is the number of expected susceptible individuals at time $t+1$ i.e. what is $\mathrm{E}[X(t+1)]$? Write it down first in terms of $P(t)$ and $X(t)$ and then substitute $P(t)$ from Q 1.2.1. Similarly, write down $\mathrm{E}[Y(t+1)]$ and $\mathrm{E}[Z(t+1)]$.

Solution:

https://drive.google.com/file/d/1IZP9L7_qqgW1GsOisNrv4nkJTdFV8JBR/view?usp=sharing

Q 1.2.3 (2 points) Assuming $\mathrm{E}[X(t)] \approx X(t)$[1], write down the expression for $X(t+1)-X(t)$? Similarly, write down $Y(t+1)-Y(t)$ and $Z(t+1)-Z(t)$.

Solution: https://drive.google.com/file/d/1U0E-5vOp95vXo0D310quZU1eSbIL7lxK/view?usp=sharing

The steps above assumed 1 unit of time. Let us change the unit of time to a small Δt instead. Hence, β should now become $\beta\Delta t$ (ditto for γ).

Q 1.2.4 (2 points) Write down $X(t+\Delta t)-X(t)$, $Y(t+\Delta t)-Y(t)$, and $Z(t+\Delta t)-Z(t)$, using your answers in Q 1.2.3 and the change mentioned above.

Q

Solution:
https://drive.google.com/file/d/14nthiwV4YvG483kah48O_aDZ1K8WGs62/view?usp=sharing

1.2.5 (3 points) Starting from your answers in Q 1.2.4, derive the standard SIR ODEequations you have seen in class. Make sure you show all the steps.

Solution: https://drive.google.com/file/d/1viyWv2rhHPyZ8_9smCaVQm0Z8-96whI7/view?usp=sharing

Q 1.3 (32 points) **Implementation and Calibration.** In this question you will implement standard SIR ODE and SIR network models.

[1]This is an example of the 'mean-field' approximation, frequently used in the analysis of non-linear models.

Q 1.3.1 (6 points) Implement the standard SIR ODE model (lecture 3, slides 18-23) in your favorite language. Your code should take in the β and γ values, and the initial fractions $S(0)$, $I(0)$, $R(0)$, and max time as input, and give the fraction of infec- tions, susceptible and recovered population (i.e. list of tuples $(t, S(t), I(t), R(t))$for t = $[0, \ldots,$ max time]) at each time-step till max time. Submit your code.

Set the values of parameters as $\beta = 0.1$, $\gamma = 0.05$ and initialize $S(0) = 0.95$, $I(0) = 0.05$, $R(0) = 0$. Show SIR curves (plot of $S(t)$, $I(t)$, $R(t)$ over time) for up tomax time = 200.

Now set $\beta = 0.05$, $\gamma = 0.1$ and plot the SIR curves. Do you notice any difference in the nature of the curves? Explain.

Note: You may use your favorite programming language like Python, R, Matlab, and Julia. Instructors are familiar with Python, and Matlab and may provide help with those. You may use basic arithmetic functions from numpy and math Python packages and packages like scipy package for integration. As a rule of thumb, make sure the packages you use only help with arithmetic operations or integration and don't solve the problem for you. If in doubt, please ask the instructors.

Solution: Q1/Q-1.3.1.py

Plot_1:https://drive.google.com/file/d/1xNocWR2yDMLkxFdjW-BRM0YeuU7sW_zZ/view?usp=sharing

Plot_2:
https://drive.google.com/file/d/17uzYkXiKmlwjek6981Ux7htMVa_IZsTd/view?usp=sharing

Yes, there's a huge difference between the two plots. For the first plot, when the beta = 0.1 and gamma = 0.05, the beta is higher than gamma. This means that the disease is more likely to spread since there's a higher probability of an infected individual infecting someone else. Furthermore, since the recovery rate is lower, individuals are more likely to infect others as well since they are less likely to recover.

For the second plot, when beta = 0.05 and gamma = 0.1, the disease spreads a lot slower because individuals who do become infected are more likely to recover, thereby reducing the number of people that these infected people infect others. Therefore, the disease quickly dies out which is what we see in the graph.

Q 1.3.2 (10 points) Implement the discrete-time SIR Network model in your favorite language. Your code should take a graph-edge-list, the total number of nodes in the graph N, the β value, the γ value, the initial fractions $S(0)$, $I(0)$, $R(0)$, and the max time as input, and give the fraction of infected nodes $I(t)$, susceptible nodes $S(t)$ and recovered nodes $R(t)$ at each time-step till max time. Use the graph-edge list from example.txt. You can set any random set of nodes to *I/R/S* states at time t=0 to reach the desired fraction. Is the network from the graph-edge list acomplete network?

Also, provide SIR plots for the network simulation given the initialization parameters from the previous question Q 1.3.1. As the SIR network model is stochastic, make sure you run the simulation *50 times*, and take the average to get$I(t)$, $S(t)$, $R(t)$. Submit the code. Do you notice any differences between the averaged results to results from a single running? Explain.

Solution: Q1/Q-1.3.2.py

Plot_1:
https://drive.google.com/file/d/1gO9lgDourreD41vX34svwoN07CvhzzsJ/view?usp=sharing

Plot_2:
https://drive.google.com/file/d/1hHxiB5tQ5QnUiMUnSFhEhHzFr7hihWko/view?usp=sharing

Plot_3(single_running1):
https://drive.google.com/file/d/1T2QM6lJe8GO6UHNmae0fwr8KgZttRUz0/view?usp=sharing

Plot_4(single_running2):

https://drive.google.com/file/d/1rOctybw0wcxjy__wlv_xNpx9bIhWVft9/view?usp=sharing

Yes, the graph-edge list is a complete network. When I ran the code with the averaged results versus a single running, I noticed the plot for the single running was a lot more variable. In other words, I noticed the plot had a lot more spontaneous drops and increases compared to the smoother curves in the averaged results. This is because the randomness of infection can play a higher factor in the shapes of the curve; averaging the results significantly reduces this factor.

Hint: You can use packages to construct graphs like the networkx Python package or SimpleGraph, LightGraph Julia packages.

Q 1.3.3 (8 points) Calibrate your SIR ODE model using the COVID mortality time series for GA from July 2021 - Nov 2021 (download from Canvas). The dataset contains the cumulative fraction of cases and deaths observed each week. You will be using deaths as ground truth $R_{observed}(0) \ldots, R_{observed}(T)$. The objective function to minimize is

$$L = \sum_{t=1}^{T} (R(t) - R_{observed}(t))^2$$

Assuming the R state corresponds to mortality (Refer to lecture 4).
To initialize the boundary conditions, we set the initial values of $I(0)$ as the fraction of cases and $R(0)$ as the fraction of deaths on the first day from the dataset. Set the initial value of $S(0)$ as $S(0) = 1 - R(0) - I(0)$. Now calibrate the SIR model to find the best set of parameters β, γ that minimizes the objective function L.

Write down the values of β, γ you find after calibration. Plot $S(t)$, $I(t)$, $R(t)$ vs *time*. Also, plot another figure showing $R_{observed}(t)$ you used to calibrate and the resultant $R(t)$ after calibration. Submit the plots.

Hint: You may use pre-built optimization functions/packages. For Python, you may consider using the scipy.optimize.minimize function from scipy.optimize import minimize. Similarly, you may use DifferentialEquations Julia library.

Solution: Q1/Q-1.3.3.py

Beta = **0.04926084359985644**

Gamma = **9.175149064429577e-06**

SIR Plot 1:

https://drive.google.com/file/d/1oGukoqyt-XQ17CCJqpeV_-PMt5s-ixW_/view?usp=sharing

R vs R_observed Plot 2:

https://drive.google.com/file/d/1n-TwWbwISyxlbH7gWsBgLzUVu_vyyTG3/view?usp=sharing

Q 1.3.4 (8 points) Run the SIR Network model on a clique of size $N = 100$. Set the β and γ values as the ones you get in Q 1.3.3. Set the max time as 200. For the simulation, set the initial number of infected individuals $I0 = \lfloor I(0) * N) \rfloor$ (where $I(0)$ is the value you set in Q1.3.3), initial number of susceptibles $S0 = \lfloor (S(0) * N) \rfloor$, and the remaining nodes as $R0$ (i.e. $R0 = N - \lfloor I(0) * N \rfloor - \lfloor S(0) * N \rfloor$).

Note that $\lfloor x \rfloor$ is the function that just rounds x to the closest integer *less than* x.

Plot the $S(t)$, $I(t)$, $R(t)$ vs t on a plot as in Q1.3.3 (submit the plot). Repeat the same steps for $N = 20$ and $N = 500$. Submit the plots. Compare the plots for different values of N and for the ODE (from Q 1.3.3). What do you observe?

Solution: Q1/Q-1.3.4.py

Plot N = 20:
https://drive.google.com/file/d/1oT_qGiDz8w7mMmuByY07eA4QohRJbazP/view?usp=sharing
Plot N = 100: https://drive.google.com/file/d/1N514ZEv77B2pVlC-RPVcjstMcaJXiNrh/view?usp=sharing
Plot N = 500:
https://drive.google.com/file/d/17trLy2eZkoi82PGgIaun00K2iW36Zb8P/view?usp=sharing

As N increases, the plot becomes less variable and begin to resemble the ODE SIR plots more.

2. (10 points) Contact networks

Suppose you are studying the spread of a rare disease among the set of people pictured in Fig 1. The contacts among these people are as depicted in the network in the figure, with a time interval on each edge showing when the period of contact occurred. We assume that the period of observation runs from time 0 to time 20.

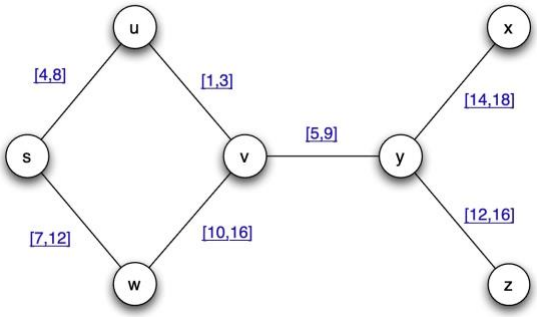

Figure 1: Contact network. Time intervals showing when the contacts occurred.

Q 2.1 (5 points) Suppose that *s* is the only individual who had the disease at time 0. Which nodes could potentially have acquired the disease by the end of the observation period, at time 20?

Solution: s, u, v, w

Q 2.2 (5 points) Suppose that you find, in fact, that all nodes have the disease at time 20. You're fairly certain that the disease couldn't have been introduced into this group from other sources, and so you suspect instead that a value you're using as the start or

end of one of the time intervals is incorrect. Can you find a single number, designating the start or end of one of the time intervals, that you could change so that in the resulting network, it's possible for the disease to have flowed from s to every other node?

Solution: **Changing the end of the time interval for edge (v, y) from a 9 to 11 would make it possible for the disease to have flowed from s to all nodes.**

3. (20 points) Calibrating IC Models

In this question, we will try out a simple way to calibrate the IC model on a graph, given some cascade traces. Recall that the IC model is just a SIR model on a network such that any infected node cures itself in 1 time-step, and each edge (v, u) may have a different constant beta probability p_{vu}. Remember that p_{vu} may not be equal to p_{uv}.

We will work with the following two datasets (download from Canvas):

1. Network: network.txt [2] Note that the graph is directed. Edge $v \rightarrow u$ is denoted as (v, u).
2. Cascades/Action log: Ratings.timed.csv [3].
   This file contains an action on each line. An action is a user rating a movie. That is,if a user v rates "The King's Speech", and then later v's friend, say u, does the same,we consider the action of rating "The King's Speech" as having propagated (cascaded) from v to u. In epidemiological terms, you can think of v 'infected' u.

In this setting, the goal of the calibration procedure is to find out the probability p_{vu} over each edge (v, u).

Q 3.1 (5 points) Let A_{v2u} be the total number of actions of node v which cascaded to node u later. An action cascades from v to u if both the following conditions satisfy:

1. v must have taken the action before u.
2. Edge (v, u) exists. In other words, v is an in-neighbor of u.

Similarly, let A_v be the total number of actions of node v in the database. Thenyou can think of node v making A_v tries to infect node u, of which only A_{v2u} tries succeeded.

https://drive.google.com/file/d/10YYaBnMa89JeF2UMhazEPLYRLGCWKhZC/view?usp=sharing

Solution:

We find the Maximum Likelihood Estimator p^*_{vu}. i.e

$$p^*_{vu} = \arg\max_{p_{vu}} L = \frac{A_{v2u}}{A_v}.$$

(Try to derive it on your own.)

Q 3.2 (15 points) Now we calibrate the IC model to estimate $p*_{vu}$ for each edge (v, u). Using the two datasets given above, write code to calculate the p^*_{vu} for each edge (v, u) of the network. The code should take in the two datasets as input, and print "v, u, p^*_{vu}" (without quotes) on each line for each edge (v, u). Submit both the code and the output you get after running it on the two datasets.

[2]The original much larger raw file we constructed this from is here: http://konect.cc/networks/flixster/

[3]The original much larger raw file we constructed this from is here: https://sites.google.com/view/mohsenjamali/flixter-data-se

Note 1: If there are two actions that happen on the same date in the cascades/action log, you can consider the one that appears earlier in the file to be temporally earlier (i.e. use line numbers to break the ties).

Note 2: There are many users u who have zero actions. In such cases, you may set $p^*_{vu} = 0$.

Note 3: There are multiple ways to do this efficiently. We will not grade on efficiency of your code. However, it might be interesting for you to hear that you can do this in < 2 scans of the action/cascades log.

Solution: **Look at file in HW1 Folder labeled "Q3.2_output.csv" and**

Q 3.3 (5 points) **[Bonus]** Let $p^*_u = \sum_v p_{uv}$ i.e. if we consider the p^*_{uv}'s as the weights on the network, then p^*u is just the weighted out-degree degree of each node u. Plot this weighted out-degree distribution. Make the y-axis log scale. Submit the plot. Whatdo you observe?

Hint: You can use the numpy function histogram() for this.

Solution:
https://drive.google.com/file/d/1NK1FIo04mlGyM3XMtppTM_Ty_eiEET8d/view?usp=sharing

4. (20 points) Zombie Apocalypse

There is a new disease outbreak in a town that turns people into zombies.

Q 4.1 (5 points) **Stopping the outbreak using branching process.** In the initial stageof the outbreak, the zombies don't harm humans, they only infect them with the disease. The town has set aside 2 million dollars to deal with this outbreak which is currently in its infancy. The public health officials have two measures to slow downthe outbreak:

1. Controlling how many people a zombie comes in contact with: If the officials spend x dollars, they expect a zombie to come in contact with $40 - \frac{x}{200,000}$people.
2. Controlling the probability of transmission: If the officials spend y dollars, they expect the probability of a zombie transmitting the disease to another person it comes in contact with is $0.04 - \frac{y}{100,000,000}$dollars.

What is the best way to allocate the 2 million dollar budget?

(Assume that most people are uninfected. Therefore, number of people an infected zombie comes in contact with is assumed to be constant.)

Hint: Refer to Lecture 5 Slides 36-41.

Solution:

Solution:
https://drive.google.com/file/d/1eEtrMWMffMao8PaHePFIH31rdU7JJiF4/view?usp=sharing

Graph Showing Solution:

https://drive.google.com/file/d/1ZIbiy90L8NjhrFei4idSZDfDlOHv9Dzu/view?usp=sharing

Q 4.2 (15 points) You failed to stop the outbreak. Now the zombie disease has turned sin- ister. The infected zombies don't merely turn humans into zombies, instead, they consume them.

Q 4.2.1 (6 points) Assume $x(t)$ is the population that is human and $y(t)$ the the population that are zombies at time t. We use the Lotka-Volterra Model (LVM) to model the

dynamics of the populations:

$$\frac{dx}{dt} = \alpha x - \beta xy \quad (1)$$

$$\frac{dy}{dt} = \gamma xy - \delta y \quad (2)$$

Similar to Q 1.3.1, implement the ODE model above that takes in parameters $\alpha, \beta, \gamma, \delta$ and initial states $x(0), y(0)$ and returns the fraction of populations $x(t), y(t)$ till time-step max_time. Submit the code for the model.
Set $\alpha = \beta = \gamma = \delta = 1$. Now set the values of initial population $x(t) = 5, y(t) = 2$ and plot the values of $x(t), y(t)$ vs. t till $T = 100$. Submit the plot. Now experiment with two other initializations of $x(0), y(0)$ for which the values of $x$and y don't change over time.

Solution: Q4/Q-4.2.1.py

Plot_1:
https://drive.google.com/file/d/13xZ0yYD47pVPZT4rOzS1WIr70izCb7Z5/view?usp=sharing

Plot_2:

https://drive.google.com/file/d/1FVIhuOtSnxbbRrVT0PLnhlrST6NRb5Fb/view?usp=sharing

Plot_3:

https://drive.google.com/file/d/10pvfH4JtpHO8Sw91xE_yZVpvLtilkwah/view?usp=sharing

Q 4.2.2 (3 points) The zombie disease has mutated further! Now you realize that there are actually two variants of zombies that are consuming humans. Assume $x(t)$ is the population that is human $y(t)$ the population that is zombie variant 1 and $z(t)$ the population that is zombie variant 2 at time t.
We model the population dynamics as a variant of LVM called LVM2:

$$\frac{dx}{dt} = \alpha x - \beta xy - \phi xz \quad (3)$$

$$\frac{dy}{dt} = \gamma xy - \delta y \quad (4)$$

$$\frac{dz}{dt} = \rho xz - \epsilon z \quad (5)$$

What are the fixed points of this LVM2 model?
Hint: You may find the fixed points by setting all partial derivatives to 0, and solving the equation pairs.

Solution:
https://drive.google.com/file/d/1bTK3isHugNf1PCGRKlFvilj_gCiAnMep/view?usp=sharing

Q 4.2.3 (5 points) Implement the model in your favorite language that takes in param- eters α, β, γ, δ, ϕ, ρ, ϵ and initial states $x(0)$, $y(0)$, $z(0)$ and returns the fraction of populations $x(t)$, $y(t)$, $z(t)$ till time-step max time. Submit the code.
Assume $\alpha = \beta = \phi = \gamma = \rho = 1$, $\delta = 1.5$ and $\epsilon = 2$. For $x(0) = 4, y(0) = 2, z(0) = 5$ plot $x(t)$, $y(t)$, $z(t)$ vs. t till $T = 100$. Also plot for other initializations of $x(0)$, $y(0)$, $z(0)$ for which they are fixed points and show that the population doesn't change over time. Submit the plots.

Solution: Q4/Q-4.2.3.py

Plot 1 (Original):
https://drive.google.com/file/d/13xZ0yYD47pVPZT4rOzS1WIr70izCb7Z5/view?usp=sharing

Plot 2 (Fixed Point 1):

https://drive.google.com/file/d/18QSl7rpma7K8zcgryyDmUWioHf3Aas43/view?usp=sharing

Plot 3 (Fixed Point 2):

https://drive.google.com/file/d/1ohmL98qqlcvTWu1WPzN3M8GaPd_3bpoR/view?usp=sharing

Plot 4 (Fixed Point 3):

https://drive.google.com/file/d/18ClGzLY9NDtumCIsPQOYuMDLFzsfV_zo/view?usp=sharing

Q 4.2.4 (1 point) Are the $LVM2$ plots periodic?

Solution: No, the LVM2 plots are not periodic since the 2nd variant completely goes away.

5. (5 points) Closing Triangles

Imagine a social network consisting of the students in this class (if x knows/is friends with y, then there is an undirected edge between x and y). Let us create a subgraph centered around you. First, list down all the people from the class you know. These students are your neighboring nodes in the graph. Now, for each of your neighbors, ask them to list down the people they know from the class. Connect these people to your subgraph as well. These are your 'friends-of-friends' (f-o-f).

Naturally, some of your f-o-f will also be your friends. These create 'triangles' centered around you in the graph.

1. Draw the constructed subgraph with the names of your friends or f-o-f on the nodes. (You can use any plotting package or submit a handwritten graph.)
2. How many triangles are you already part of?
3. How many incomplete triangles are you part of? These are exactly the people who are friends of your friends but not (yet) yours aka 'potential' friends :-)

Hint: You have to name people in the constructed graph. Feel free to use any method to get this answer: Piazza post, Face-to-face meetings, Phone calls, Social Media, Zoom/Teams chats.

Solution:
https://drive.google.com/file/d/1YuS5q9RRc6Wfv3V4688nJF1XGaR7X0Rn/view?usp=sharing