Mini-project 1: Contagion

CX 4230, Spring 2024[1]

Spring 2024: Due Feb 16 at 11:59 pm

[1] Last updated: `Mon Jan 29 14:29:16 EST 2024`.

In this mini-project, you will simulate two infectious disease models, one that we give you and one that you devise on your own. This assignment is inspired by Chapter 21 of the book, *Scientific Computing with Case Studies* ("SCCS"), by Dianne O'Leary, which should serve as a useful reference.[2]

[2] Dianne O'Leary. *Scientific computing with case studies*. SIAM, 2009. DOI: 10.1137/9780898717723. Available via the GT Library: [Link]

Here are the basic ground rules:

- You may work individually or in pairs. See section 2.
- You can implement your solution in any programming language.
- You will submit a written write-up (PDF file) *and* your source code. See section 3.
- The assignment is due on Friday, February 16, at 11:59 pm, but see section 3 for the late policy.

1 *Your Assignment: Implement a "Well-Mixed" SIR Model*

Consider the model of a disease presented as "Model 1" in SCCS Chapter 21. In that model, there are three types of people:

1. *susceptible* people, who have never had the disease;
2. *infected* people, who currently have the disease;
3. and *recovered* people, who had the disease once but no longer have it.

The disease can only spread from infected people to susceptible people. An infected person stays sick for some period, but then recovers; once they have recovered, they can never get sick again.

Let's assume the behavior of the disease is governed by the following mathematical model.[3] Denote time by a continuous variable $t \geq 0$, and let the state of the system be represented by the three-dimensional vector,

[3] See the explanation in SCCS Chapter 21 for a more detailed explanation.

$$\vec{x} = \vec{x}(t) = \begin{bmatrix} S(t) \\ I(t) \\ R(t) \end{bmatrix}, \tag{1}$$

where $S(t)$, $I(t)$, and $R(t)$ denote the fractions of the population who are susceptible, infected, or recovered, respectively, at time t. Since we will interpret these state variables as fractions, their domain is $0 \leq S, I, R \leq 1$. Further assume no one is born or dies, so that

$S(t) + I(t) + R(t) = 1$ for all time. Thus, this constraint expresses a *conservation law* for this model.

Suppose the state $\vec{x}$ evolves as an ordinary differential equation,

$$D\vec{x} = \vec{f}(\vec{x}), \tag{2}$$

where $\vec{f}(\vec{x})$ is the vector function,

$$\vec{f}(\vec{x}) = \begin{bmatrix} -\tau S(t)I(t) \\ \tau S(t)I(t) - \dfrac{I(t)}{\kappa} \\ \dfrac{I(t)}{\kappa} \end{bmatrix}. \tag{3}$$

The parameter $\tau \geq 0$ measures how quickly the disease can spread, with higher values corresponding to faster rates of spread. The parameter $\kappa > 0$ measures how quickly an infected person recovers, with higher values corresponding to slower (longer) recovery times.

1.1 *Question 1.1 (2 points)*

We claim that the original model eq. (3), can be simplified to the following equivalent "hat" system,

$$\hat{D}\begin{bmatrix} \hat{S} \\ \hat{I} \end{bmatrix} = \begin{bmatrix} -\hat{S}\hat{I} \\ (\hat{S}-1)\hat{I} \end{bmatrix}, \tag{4}$$

for an appropriate definition of the "hat" variables and operator ($\hat{S}$, $\hat{I}$, and $\hat{D} = d/d\hat{t}$). Show how to derive eq. (4) from eq. (3) using variable substitution and the conservation law. What is the domain of $\hat{S}$ and $\hat{I}$?[4]

[4] Recall that the original system has the domain of S and I to be $[0, 1]$. What is the domain of your new, rescaled variables?

> The equations of state, eq. (4), are remarkable in that they have *no* explicit model parameters. However, such parameters *should* make an appearance; in this case, you should see them "rëappear" in the domain constraints (the last part of this question).

1.2 *Question 1.2 (3 points)*

What are the fixed points of the simplified system, eq. (4)? Classify their stability.

1.3 *Question 1.3 (10 points)*

Similar to "Challenge 21.1" of SCCS, implement a computer simulation of this model. You can use any off-the-shelf ODE solver for this task.[5]

[5] Challenge 21.1 suggests using MATLAB's `ode23s`, but you can use any reasonable equivalent. For example, if you are using Python, you can use Scipy's `integrate.solve_ivp` with its default solver, `'RK45'`.

Use your simulator to conduct the following experiment. **Note:** This simulation and the parameters below are expressed in terms of the *original* system, eq. (3). Your simulation may use either the original system or the simplified one, eq. (4); if you use the latter, remember that you should present all results in terms of the original system, not the rescaled one.

- Let the initial conditions be $I(0) = 0.01$ (initial infected population of 1%), $S(0) = 0.99$, and $R(0) = 0$.
- Let $\tau = 0.8$ and $\kappa = 4$.
- Run a simulation until $I(t) < 10^{-4}$, the "stopping condition" for this experiment. At what time t does this condition occur? Report your stopping time results to two digits after the decimal point.[6] Create a plot that shows how $S(t)$, $I(t)$, and $R(t)$ vary in time. A sample plot appears in fig. 1.
- Run another simulation for $\tau = 0.4$ and $\kappa = 4$ with the same stopping condition. Again, note the time t at which the stopping condition occurs. Create another plot showing the state variables over time.
- Run a third simulation for $\tau = 0.8$ and $\kappa = 8$, note the stopping time, and create a third plot.
- Summarize what you observe across the three plots. For example, how do the stopping times compare? If any curves exhibit peaks, how do the peaks change (e.g., how do they shift in time and magnitude)?
- Suppose you were speaking to someone in public policy who does not understand the math. How would you help them interpret your findings qualitatively? That is, now write up a summary of your findings as if you were speaking to this public policy expert.

[6] Since you often do not know how long to run a simulation, many off-the-shelf ODE solvers allow you to define "events," that is, a stopping criteria like this one. So try to use such a facility if your ODE solver has one. If your solver does not have one, explain what you did to determine the stopping time.

1.4 *Question 1.4 (10 points)*

Conduct the following additional experiment using your simulator from Question 1.2. Try a variety of values of $\tau \in (0, 4]$ and $\kappa \in [1, 5]$.[7] For each combination of τ and κ, use your simulation to determine at what time t the number of infections falls below 10^{-4}. Create a 2-D contour plot or heatmap of t as a function of τ and κ and summarize in words what you observe.

[7] Say, 8–10 values for each, or a total of 64–100 combinations.

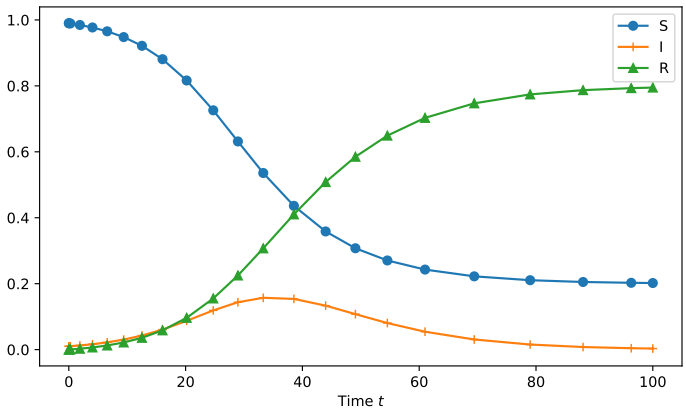

Figure 1: A sample plot of the state variables, $S(t)$, $I(t)$, and $R(t)$, in the unvaccinated model for some choice of τ and κ (not shown).

1.5 *Question 1.5 (10 points; pairs only)*

Suppose there is a vaccine that can be given to *susceptible* people. A susceptible person who receives a vaccine can no longer become infected. However, this vaccine is very expensive, so you cannot vaccinate all susceptible people.

To incorporate vaccinations, let's introduce a fourth state variable, $V = V(t)$, which is the fraction of the population who are vaccinated at time t. The new conservation law is $S + I + R + V = 1$, and we can model a vaccination scenario by the ODE system,

$$D\begin{bmatrix} S \\ I \\ R \\ V \end{bmatrix} \equiv D\vec{x} = \vec{f}(\vec{x}) \equiv \begin{bmatrix} -\tau SI - g(\vec{x}) \\ \tau SI - \frac{I}{\kappa} \\ \frac{I}{\kappa} \\ g(\vec{x}) \end{bmatrix}, \tag{5}$$

where $g(\vec{x})$ is a function that describes some vaccination policy. This function "moves" people out of the susceptible population and into the vaccinated population, where they remain.

Consider the following two different vaccination policies:

- **Policy 1:** $g(\vec{x}) = g_1(\vec{x}) \equiv \nu_1 S$ — Under this policy, we simply vaccinate at a rate proportional to the current population of susceptible individuals: the more there are, the more we try to vaccinate.
- **Policy 2:** $g(\vec{x}) = g_2(\vec{x}) \equiv \nu_2 \frac{SI}{S+I}$ — This policy is suggested in the last exercise of SCCS Chapter 21.

First, try to interpret Policy 2 in your own words. What does it seem to be trying to do compared the Policy 1?

Next, create and run simulations to compare the two models. For your simulations, you should sweep a space of (τ, κ) values as you

did in Question 1.4, but now also consider a few possible values of ν and see what happens under vaccination. Does one work better than the other? Or are there tradeoffs among, for example, the total number of people who end up being vaccinated (recall that vaccinations are "expensive"), the number who become infected, and the time to reach a steady state? Be sure to clearly show representative results and summarize your findings.

2 *Teaming*

You may work individually or in pairs. Teams of two have additional requirements for the assignment, as noted above.

To "declare" your team, do the following:

- Visit the Canvas "People" page for Mini-project 1 ("MP1").[8]
- Create a new group, even if you are choosing to work individually. If you are working in a team of two, invite your other team member or join such an invitation.

[8] Link: https://gatech.instructure.com/courses/372568/groups#tab-42861

3 *How to Submit*

Create a **private** git code repository at `github.gatech.edu` (*not* `github.com`!) For teams, only one person needs to create the repository. Place both your code and a PDF report summarizing your results in this repo. You'll submit the URL to this repository on the page for this assignment in Canvas. In the repository itself, create a `README.md` file that tells us a) your individual or team number and member(s) of the team; b) which file is your PDF report; and c) what the organization of your code is (so we can inspect and evaluate it).

Important! Since your repository will be private, please be sure to add *everyone* from the teaching staff to your repo: the two TAs, Rahul Komatineni (`rkomatineni6`) and Youjie Zhang (`yzhang3988`), and Prof. Vuduc (`rvuduc3`). Otherwise, we will not be able to see and grade your submission and you will get a zero.

Late submissions. You get an automatic extension through the weekend: to submit without penalty, submit by Sunday, February 18, at 11:59 pm. There is a 10% penalty if you submit up to one day late (Monday, February 19), a 25% penalty if you submit two days late (Tuesday, February 20), and 50% penalty if you submit three days late (Wednesday, February 21). No assignments will be accepted after that time.

References

Dianne O'Leary. *Scientific computing with case studies*. SIAM, 2009. DOI: 10.1137/9780898717723. Available via the GT Library: [Link].