ESE 105 FL 2023: Case Study 2: Modeling COVID dynamics in populations

Part I Due: 10/27/2023, At Lab
Overall Due: 11/03/2023, 4:59PM

Preface

This case study is meant to allow you to work on a real engineering problem. There is no ‘right’ approach, nor is there a ‘secret trick’ that will achieve perfect results. The goal is not to create an ideal model, but rather to understand the tools at your disposal and their capability. Feel free to be creative, but also try and ground any decisions you make in terms of fundamental concepts that we have discussed in class. Using ‘black box’ techniques that you find online, without any grounding or justification, is not in the spirit of this case study.

Milestones

To be on track and working at a steady pace, you should hit these milestones:

- Part I: Complete by Tue 10/24. **Note that Part I is due by the time of your lab!**
- Part II: Complete by Fri 10/27
- Part III: Complete by Mon 10/30
- Part IV: Complete by Wed 11/01
- Writeup and Deliverables: Complete by 11/3

1 Introduction

In this case study you will use the skills and methods you have learned in linear algebra and MATLAB to simulate dynamical systems and use them for engineering design. This case study will link to the ESE topic areas of dynamical systems modeling and control engineering.

A dynamical system model describes the evolution of a system as a function of time. Many dynamical systems models are based on fundamental laws of physics (e.g., a differential equation that describes the velocity of a car), whereas others that are more phenomenological (i.e., they are based on observations and/or high-level intuition). In this case study we will be working with phenomenological models of epidemic spread in populations and networks.

1. Fit this model to data from the ongoing COVID19 pandemic. Specifically, you will attempt to fit your model to disease progression in St. Louis, MO. You will then make analytical inferences regarding how the disease progressed differently during different phases or waves of the pandemic.
2. Use your generalized model to perform a retrospective ‘what-if?’ analysis to posit how the trajectory of the pandemic may have changed given different intervention strategies.
3. Design and challenge: Generalize the SIR model by adding new states to account for vaccinated individuals. Use this to infer vaccination rates and vaccinated proportions from case and death data alone.

2 Part I: SIRD Modeling

Implement the SIRD model described in Section 9.3 of the book and become familiar and comfortable with the model. This model describes the evolution of a disease within a population. Once implemented, you should modify the model parameters to examine how the model behaves for different imagined scenarios.

2.1 Action item for Part I:

1. Write MATLAB code/scripts that simulate this model from scratch. Explore how changing the parameters of the model affects the output. How does the output of the model tend to converge over time? Produce plots that are similar to those depicted in the book. Use Matrix and Vector operations within your code!! Do not use pre-packaged MATLAB functions such as 'lsim' at this stage.

2. Modify your model to implement a scenario in which re-infections are possible. Show an example of model outputs in this case, and interpret the observed behavior.

3. Now, examine the script *'base_sir.m'*, which implements the model from Section 9.3 using the MATLAB function *'lsim'*. Use MATLAB help to learn more about *'ss'* and *'lsim'* and what is happening in this script. Here are some key points:

   - *'lsim'* allows for simulation of a general class of linear dynamical systems, including ones that have external inputs.
   - The script first sets up the model using the *'ss'* function. It uses the specification of the *'A'* matrix and sets a number of parameters. Read about the *'ss'* function to learn more. Note especially that we are working in the space of discrete-time models, where time advances in steps (as opposed to continuous time models, which are defined in terms of differential equations).
   - Experiment for yourself with base_sir.m and the ss and lsim functions. Compare the output of using these functions with the outcome of your from-scratch effort in Part (i).

3 Part II: Fit the model to real data

3.1 Major ideas and methods

3.1.1 COVID Data

Load the MATLAB file *'COVID_STL.mat'*, which contains COVID data for St. Louis city and county. Familiarize yourself with this data (this is very similar to what you worked with in your previous MATLAB homework). In particular, the relevant variables are:

1. cases_STL – vector containing **cumulative** COVID cases for St. Louis

2. deaths_STL – vector containing **cumulative** COVID deaths for St. Louis

3. dates - vector containing the date corresponding to each entry of cases_STL and deaths_STL

4. POP_STL - total population of St. Louis city and county

3.1.2 Tuning your model to explain actual COVID19 data

Once you are comfortable with *'base_sir.m'*, you move on to use the model to explain actual data. Here, your goal is to find values for the SIRD model rate-constants (i.e., infection rate, fatality rate, recovery rate) and initial conditions (initial values of susceptible, infected, recovered and deceased population fractions) that best predict the actual case data over certain time-ranges of the pandemic. For example, Figure 1 shows the output of a model that is tuned to predict the first 100 days of case data from St. Louis Metro.

There are several ways to tune model parameters. The most basic is the tune by hand, whereby you manually adjust the parameters in order to reduce the error between the model output and the data. When hand-tuning, make sure you obey the contraints of the model (e.g., no negative rates). In later courses, you will learn about ways to 'auto-tune' parameters, via optimization. If you wish to explore parameter optimization in MATLAB, you can look into the function '*fmincon.*'

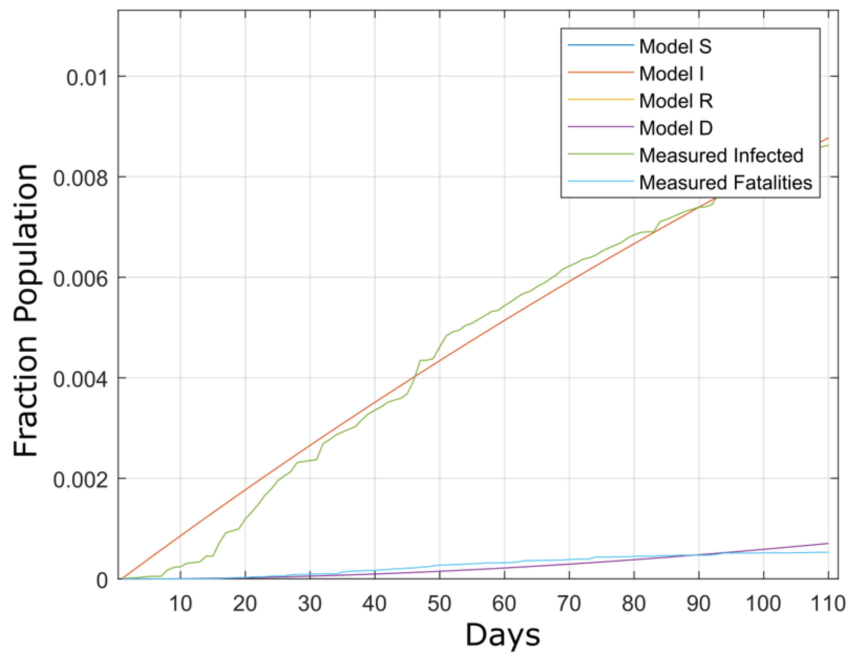

Figure 1: Example of model fit to the first 100 days of data from St. Louis Metro

3.1.3 Predicting the effects of a policy intervention

Your model can be used to explore the efficacy of different interventions, e.g., mask mandates, lockdowns, vaccinations, etc. Once you have obtained your model, perform a 'what if?' analysis by implementing a policy of your design and deploying it at a strategic date. You will have to translate the policy (e.g., mask mandates) into a mathematical effect within the model. Be prepared to describe and justify this process.

3.2 Action items for Part II:

1. Many experts speak of different phases or waves of the disease. Consider the phases of the pandemic roughly from 6/30/21–10/26/21 and 10/27/21–3/22/22. These time periods were associated with the 'Delta' and 'Omicron' variants, respectively. Tune different models for these different phases by isolating those ranges of time (approximately) and then hand-tuning. Compare/contrast the different tuned models. Do your models speak to differences in viral propagation dynamics between these phases?

2. The above phases were also times when various policies around COVID were changing. Do your model fits suggest a change in policies between the waves?

3. Related to the above, now use your model to perform a 'what-if' type of analysis. Try to design a policy that by your design would have resulted in a 25% reduction in cases and deaths during the Omicron wave. Consider the following:

   (a) What is your policy? How is it implemented mathematically in the model? Does it achieve the desired effect?

   (b) Is your policy feasible? In other words, will the societal costs be too great for this policy to be worthwhile?

4 Part III: The effects of travel

4.1 Key ideas and methods

4.1.1 Adding inputs to the model

The SIRD model in its basic form is an example of a 'closed' system, meaning the overall population (sum of S,I,R,D) does not increase or decrease. However, this assumption becomes problematic if we want to consider the effects of travel. How would we, for example, model the interaction of two distinct populations?

4.2 Action items:

1. Suppose you have two populations (e.g., two metro areas) interacting. Extend your model to describe the viral dynamics within these populations, as well as travel *across* these populations. What are your assumptions? How do these correspond to your modeling choices? What would the 'combined' model of the two interacting populations look like? Start by writing some equations, then simulate! Perform a what-if study to analyze how the rates of travel between these populations would alter their respective case trajectories.

2. Now, perform a what-if policy analysis by examining how different restrictions regarding travel might have impacted case dynamics at different phases of the pandemic. Choose at least two geographic regions, and at least two different phases (e.g., the Delta and Omicron waves). Start by simulating under a baseline set of assumptions regarding travel, then gradually implement your policy. Discuss your assumptions/model, and the societal interpretations or implications of the policy.

5 Part IV: Modeling vaccination and competition challenge

In this final part of the Case Study, you will generalize the model to be able to account for vaccinations.

5.1 Competition scenario

We have provided you with mock data (in *'mockdata.mat'*) describing the fraction of a population that is infected over 400 simulated days. The mock data also contains the fraction of a population that is deceased as a function of days. In this data, you know that vaccinations did not exist prior to a certain day, and were rolled out after that day. Your goal is to infer a more complete breakdown of the population over time, including:

1. When vaccines were rolled out

2. Fraction of population that is vaccinated over time

3. Portion of the population experiencing a breakthrough infection over time

For the purposes of this model generalization, you should assume that there is a certain rate of 'breakthrough infections', i.e., when vaccinated individuals become infected. In the mock data, this rate is fixed (after day 100).

5.2 Action items:

1. Add additional state variable(s) to the model to account for vaccinations (in the base model, there are four state variables: S, I, R, D).

2. Augment the relevant transition matrices by adding new rate constants between states, as appropriate.

3. Use your model to try and infer the 1) fraction of population that has been vaccinated and 2) fraction of the population experiencing a breakthrough infection as a function of time. Employ parameter optimization techniques, such as developed in Part II, to your advantage.

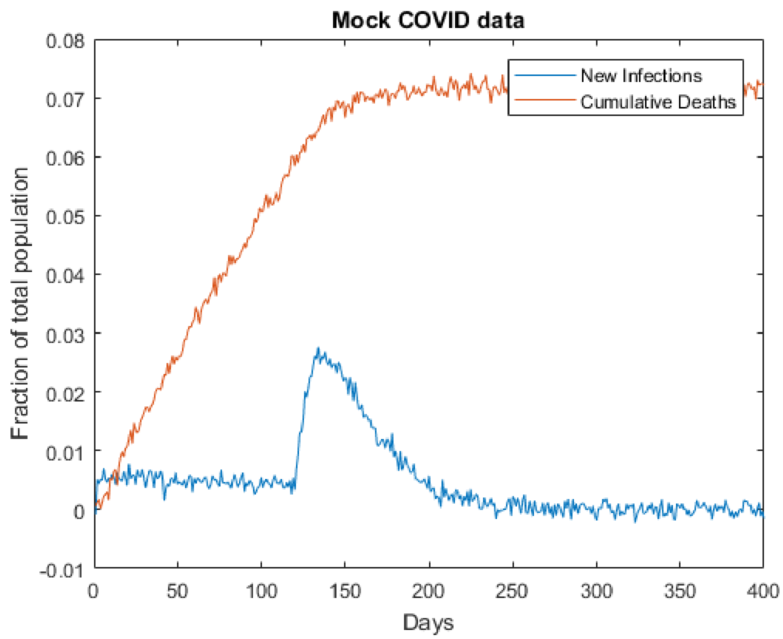

Figure 2: Mock COVID data for a simulated city

6 What to Turn In

1. A completed and signed honor code indicating your full and complete participation in the case study.
2. Your built-up and published MATLAB files, including at minimum:
   (a) a script called *'part1.m'* and published file *'part1.pdf'* that produces any figures supporting Part I and its action items.
   (b) a script called *'part2.m'* and published file *'part2.pdf'* that produces any figures supporting Part II and its action items.
   (c) a script called *'part3.m'* and published file *'part3.pdf'* that produces any figures supporting Part III and its action items.
   (d) a script called *'part4.m'* and published file *'part4.pdf'* that produces any figures supporting Part IV and its action items.

   Use the MATLAB publish command as appropriate to highlight key steps or outputs, similar to what you have done on MATLAB homeworks. Make your own decisions regarding what steps are most salient and need to be highlighted, as well as what figures are needed to support your results. Note that you may submit additional files upon which the above scripts depend.
3. A MATLAB data file (i.e., .mat file) called *competition.mat* that contains the following:
   (a) A 365-dimensional array *'vaxpop'* that contains your inferred fraction of the population that **has been** vaccinated as a function of days.
   (b) A 365-dimensional array *'vaxbreak'* that contains your inferred fraction of the population that is experiencing a breakthrough infection as a function of days.
4. A 4-6 page report illustrating your simulations and design choices. Use the templates provided with the case study materials. Make sure that your report clearly states/presents:
   (a) All modeling and design choices, especially pertaining to the model generalization.

(b) Outcomes and discussion associated with each part, including any inferences made about the actual COVID trends in St. Louis based on your model fitting.

(c) Conclusions about this modeling framework and design strategy, including limitations, **based on your quantitative findings presented in the report**.

(d) **Use plots or diagrams to support your claims and arguments.** Make sure all plots, axes and axes labels are legible. Use the figure export tools within MATLAB (e.g., MATLAB function "exportgraphics()"), and avoid the use of screen capture tools. Your report will use the IEEE 2-column format. Templates for MS Word and Latex are provided.

(e) **Important:** Claims that are unsupported or unsubstantiated by figures or published MATLAB code will be heavily scrutinized.

7 Rubric

- Correctness of MATLAB code - 40%
  - Implementation of SIRD model
  - Fitting of COVID data
  - Generalization of SIRD model
  - Items in *competition.mat*
- Presentation - 20%
  - Plots are easy to read and interpret, with appropriate font sizes, line widths, axis labels, etc.
  - Report should be well-organized, concise, and clearly written.
- Programming style - 20%
- Study design - 20%
  - Report addresses specific considerations noted above.
  - Methodology is presented such that an informed reader would be able to to reproduce the findings.

8 Data sources

- Annual Resident Population Estimates (CO-EST2021-ALLDATA), United States Census Bureau.
- Coronavirus (Covid-19) Data in the United States. Data from The New York Times, based on reports from state and local health agencies.