1. Introduction of the Flatten-the-Curve (FC) controller

The idea behind the FC controller is to allow the disease to spread, as long as the peak of the pandemic curve (i.e., the daily number of infectious people over time) does not exceed what the local hospital capacity can accommodate. The controller is implemented in the following two steps.

Step 1: A pandemic curve is simulated. The local hospital capacity would be just enough to meet the hospitalization needs from the infectious people at the peak of the simulated pandemic curve. We call this simulated pandemic curve the **target curve**.

Step 2: The transmission rate (i.e., the number of susceptible people each infectious person could infect each day) is recommended on a daily basis. To do so, the actual number of infectious people on that day is compared to the number on the target curve. If the actual number is below the target curve, the transmission rate on the next day should be increased; otherwise, it should be decreased. The goal of adjusting the transmission rate is to make the daily number of infectious people stay close to the target curve.

2. Description of the code scripts

In a real-world setting, data on the actual daily number of infectious people often comes with biases, most notably the underreporting issue, which means the number of infectious people that reported is smaller than the true number of infectious people. Around the severity of the underreporting issue and whether the controller attempts to correct the underreporting issue, we designed 7 scenarios, each implemented by one code script.

- Script “simulation_baseline_and_sce2_075.R”: This script implements two scenarios. The first scenario is the baseline scenario, where there is no underreporting issue. The second scenario is where the data is underreported, but the controller corrects the data to be the true number of infectious people. From controller input perspective, these two scenarios are identical and thus implemented using the same script.
- Script “simulation_sce1.R”: This script implements the scenario where the data is underreported and the controller is not correcting the data (and thus using the data directly for **Step 2**). In terms of the severity of underreporting, it is assumed that only 15% of true number of infectious people on each day was reported in the data.
- Script “simulation_sce2_005.R”: This script implements the scenario where the data is underreported (with the same 15% reporting percent as in “simulation_sce1.R”), and the controller mistakenly assumes the reporting percent to be 5%, and thus upscale all reported number of infectious people in the data by 1/0.05. Consequently, the upscaled number of infectious people is used in **Step 2**.
- Scripts “simulation_sce2_010.R”, “simulation_sce2_025.R” and “simulation_sce2_05.R”: These 3 script implements scenarios similar to the one by “simulation_sce2_005.R”, where there is underreporting issue and the controller attempts to correct it. Their differences are in the reporting percents assumed by the controller and thus the upscaling factors applied to the reported number of infectious people. In the three scenarios corresponding to the three scripts, the controller assumes the reporting percent to be 10%, 25% and 50%, and thus applies the upscaling factor of 1/0.1, 1/0.25 and 1/0.5, respectively.

3. Guide for running the code scripts

The 6 code scripts have similar structures and organizations. In the remaining of the tutorial, the script "simulation_sce2_005.R" will be used to explain what each section in the script does and how to run them.

3.1. Prerequisite

The scripts were developed using the R programming language of version 4.0.1. A more recent version of R will be compatible.

The R libraries required for running the scripts include "lamW" and "deSolve". The "lamW" library contains the Lambert-W function, which is used in **Step 1** to calculate the transmission rate for generating the target curve. The "deSolve" library is used for solving and forecasting in differential equations, which is necessary to obtaining the target curve in **Step 1**.

No input file is needed for running each code script.

3.2 Note on each section of the code scripts (using "simulation_sce2_005.R" as example)

Lines 4 to 9 define the parameters in the study setup. These parameters include the population in the area (N), the number of infectious people the local hospital capacity can accommodate (I_max), the initial number of the infectious people (I_0), the initial number of removed people (R_0), and the removal rate (gamma_SIR). Note that I_0 is the true number of infectious people, not that reported in the data.

Lines 10 to 40 implements **Step 1**. They first calculate the transmission rate (beta_obj) that will be used for generating the target curve. The initial infectious and removed numbers also serve as input to generating the target curve, as indicated by line 14. Lines 16 to 27 define an SIR model using its continuous differential equation formulation. And lines 29 to 32 run a simulation of the SIR model for 730 days since the initial date. The simulation produces the target curve (the daily number of infectious people over the 730 days), which is outputted to a plot in lines 34 to 40.

Lines 46 to 77 implements **Step 2** where there is no data issue (i.e., either underreporting or correction to it). The initial conditions are first defined in lines 46 to 49. Lines 51 to 55 defines (empty) output variables that will be used to store the daily recommended transmission rate (beta_t), daily number of susceptible people (S_t), daily number of infectious people (I_t) and daily number of removed people (R_t), during the process of running **Step 2**. Lines 56 to 77 run the **Step 2**: for every day of the 730 days, the controller calculates the recommended transmission rate by comparing the infectious number with that on the target curve, as in line 57; then it runs a simulation of the SIR model for 1 day using the recommended transmission rate and the susceptible, infectious and removed numbers on the current day to get the susceptible, infectious and removed numbers for the next day; and for the next day, this process repeats and so the iteration goes on until it covers all 730 days.

Line 79 to 96 summarize and output the Step 2 results as csv files and figures.

Lines 103 to 136 also implement **Step 2**, in situations where data issues (underreporting or its correction) exist. The key difference from the previous section on line 114, where the reporting percentage (15% or 0.15) and the upscaling factor for correction (1/0.75) are applied to the true number of infectious people. Other parts of this section are the same as the previous section, and the step outputs are stored and visualized in lines 138 to 159.