Data Analytics - Assignment #5

Nishant Kumar (Sr.No: 21495)

October 2023

Covid-19 Modelling

1 Problem Statement

We're using a model to predict COVID-19 cases based on the number of people getting their first vaccine dose and how long immunity lasts. We adjust some key values in the model to make it match the actual COVID cases between March 16, 2021, and April 26, 2021. Then, we use these adjusted values to make predictions about future COVID cases and study how the contact rate (how easily the virus spreads) affects these predictions.

2 Methodology

Formulation of the Problem:

We Implement the following equations given below as part of the SEIRV model:

$$\Delta S(t) = -\beta(t)S(t)\frac{I(t)}{N} - \epsilon\Delta V(t) + \Delta W(t) \tag{1}$$

$$\Delta E(t) = \beta(t)S(t)\frac{I(t)}{N} - \alpha E(t) \tag{2}$$

$$\Delta I(t) = \alpha E(t) - \gamma I(t) \tag{3}$$

$$\Delta R(t) = \gamma I(t) + \epsilon\Delta V(t) - \Delta W(t) \tag{4}$$

We have some fixed numbers:

- It takes an average of 5.8 days for someone to show symptoms after getting infected (mean incubation period).
- It takes an average of 5 days for someone to recover from the illness (mean recovery period).
- The vaccine we're using is about 66% effective (vaccine efficacy).
- The total number of people in our population is 70 million (N). These are the important details we're working with.

We need to make sure that when we start a certain process:

1. The initial amount (R0) should be between 15.6% and 36% of the total number of things we're dealing with (like people or objects).

2. The initial rate (CIR0) should be between 12.0 and 30.0.

let's simplify the model for immunity waning and the CIR equation:

1. From March 16, 2021, to April 15, 2021, our immunity decreases by a fixed amount, which is R(0) divided by 30.
2. After September 11, 2021, our immunity changes based on two things: the change in the COVID-19 infection rate (Delta R) and the change in vaccinations (Delta V) that happened 180 days ago. We also add a small random factor (Epsilon) to this change.

We also have another measure called CIR, which depends on how long it's been since a certain date (t0) and how long it's been since the current date (t).

We define CIR(t) = CIR(0) * T(t0) / T(t)

Pre-Processing:
I've made some changes to the dataset to make it ready for solving the problem.

1. After I open the CSV data file and convert it into a Pandas table, I only keep the parts that have information about "Date," "Confirmed" cases, "Tested" cases, and the number of "First Dose Administered." I get rid of the other parts because I don't need them to solve the problem at hand.
2. I'm making the numbers in the "Confirmed," "Tested," and "First Dose Administered" columns show how much happens each day instead of the total amount.
3. I calculate a weekly average for three specific columns of data, starting from March 16, 2021, and continuing until the end of the data. To do this, I use information from March 9, 2021, as part of the calculation.
4. I'm going to estimate the number of tests conducted until December 31st by using the latest data available up to that date.
5. I'm estimating the number of people who will receive their first COVID-19 vaccine dose by assuming that 200,000 vaccinations will be given every day from April 27, 2021, to December 31, 2021.

Compute Loss:

1. It takes some settings for a mathematical model as input.
2. It uses those settings to create a sequence of numbers representing something changing over time, like daily COVID-19 cases, using the "generateTimeSeries" function.
3. It then calculates the average of these numbers over a 7-day period to smooth out fluctuations.
4. It compares this smoothed sequence with the actual number of cases, like the real data we have.
5. It calculates a measure of how different the generated sequence is from the actual data. This measure is called "loss," and it tells us how well the model is doing at simulating the real-world situation.

$$l(p) = \frac{1}{42} \sum_{t=March16}^{April26} (\log(c^{-}(t)) - \log(\alpha e^{-}(t)))^2$$

Compute Gradient:

1. It takes some values that describe a mathematical model as input.
2. Inside the function, it uses another function called "computeLoss" to calculate a measure of how well the model works.
3. Then, it tries to figure out how sensitive the model's performance is to changes in the input values. To do this, it makes small changes (perturbations) to a few of the input values: Beta by 0.01, CIR0 by 0.1, and S0, E0, I0, R0 by 1.
4. It checks how these small changes affect the model's performance.
5. Finally, it gives back the results of these checks as a set of numbers in a format that can be used by the main part of the program.

In simpler terms, the function helps us see how well a model can predict real data by generating its own data and comparing it to what actually happened. The "loss" is like a score that tells us how good or bad the model is at making predictions.

3 Result:

After making some adjustments to the model's settings, it now produces very low error (less than 0.01).

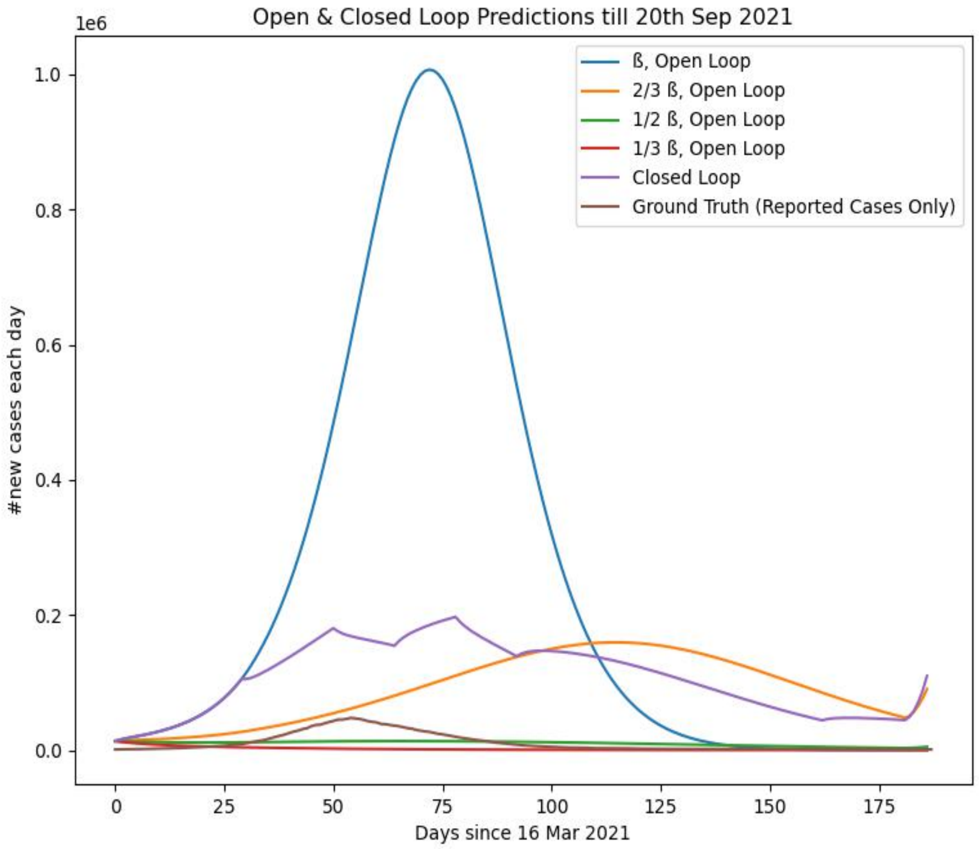
a. $\beta = 0.449722551$
b. $S_0 = 48,999,999.9$ (49 million)
c. $E_0 = 76,999.9180$ (77 thousand)
d. $I_0 = 76,999.9182$ (77 thousand)
e. $R_0 = 20,852,999.9$ (20.8 million)
f. $CIR_0 = 12.8716990$

The loss for those settings is **0.0029778394997330934**, which is smaller than the threshold of 0.01.

I've used certain settings to make predictions about what might happen in the future until December 31st. I used different values for Beta (which is a control parameter) in both open and closed-loop systems. As requested, I've also created graphs to show the results up to September 20th and compared my predictions to the actual reported cases of something (not specified in the text).

4 Plots and Observations

The first graph below shows a comparison between two methods for predicting future daily average new cases of something. One method is called "open-loop," and the other is called "closed-loop," and they use different Beta values. The graph also includes the actual number of reported cases (what really happened) up to September 20th.



Observations:

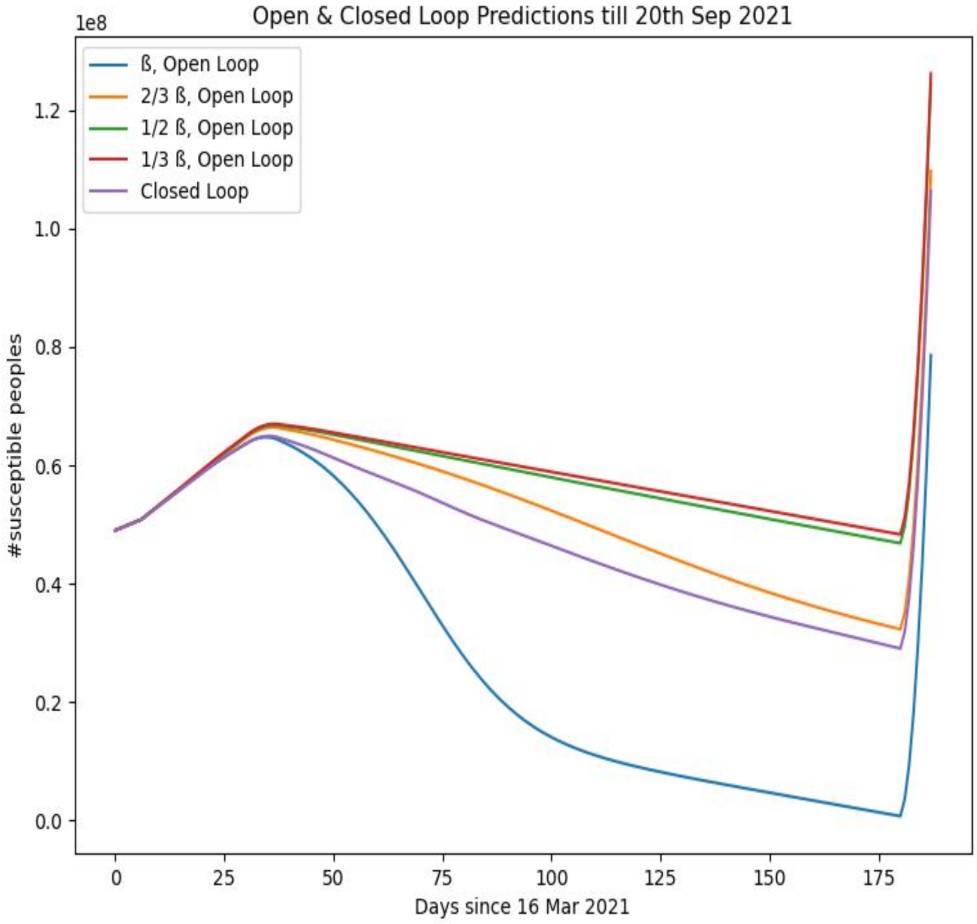
1. As we anticipated, the number of new daily cases is highest for BETA, followed by 2/3 BETA, 1/2 BETA, and 1/3 BETA. The graph for Closed-Loop control is represented in Purple color above. When we have higher BETA values, it means that because people are coming into contact more often, more individuals get infected. However, the curve also drops more rapidly because the number of people who can still get infected decreases quickly when BETA is high.

2. The Closed-Loop looks tough and has sharp edges because it changes its BETA values dynamically.

3. The ”ground truth,” which shows the actual or real numbers, is marked in brown on the graph. It’s clear that the ground truth is lower than most predictions. This is because the number of reported cases is usually much lower than the true number, especially during the worst part of a pandemic.

4. In situations where we have 1/2 or 1/3 of the usual disease transmission rate (contact rates), we can see that the number of new cases drops significantly and gradually reaches zero. This means that the disease isn’t spreading from person to person very quickly, and eventually, the pandemic fades away because it’s not easily passing from one person to another.

The second graph, shown below, displays how the number of people who could catch a disease changed up until September 20, 2021.



Observation:

When BETA is high (like in the blue graph), the virus spreads fast and infects many people rapidly. As a result, there are fewer people left who haven't been exposed to the virus (the susceptible population) because they get infected quickly. On the other hand, when BETA is low (like in the orange, red, and green graphs), the virus spreads more slowly, so there are more people who haven't been exposed to it for a longer time.