Data Analytics – Covid 19 Modelling
Assignment – 5
<u>Report</u>

By: - Pratyush Gauri Sr. No: - 20227

In this assignment, we have implemented the SEIRV model to predict the COVID19 cases. The parameters [$\beta, S_0, E_0, I_0, R_0, CIR_0$] are updated according to cases between 16th March 2021 and 26th April 2021. These parameters are used for future predictions and analysis for different contact rate.

The methodology is explained as follows: -

FORMULATING SEIRV MODEL

The equations used to create this model is given as :-

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

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

$$\Delta I(t) = \alpha E(t) - \gamma I(t)$$

$$\Delta R(t) = \gamma I(t) + \epsilon \Delta V(t) - \Delta W(t$$

Given are :- mean incubation period $[(\alpha)^{-1}] = 5.8$ days
mean recovery period $[(\gamma)^{-1}] = 5$ days
vaccine efficacy $(\epsilon) = 0.66$
total population $(N) = 70$ million

→ $R(0)$ is between 15.6% and 36% of the population
→ $CIR(0)$ is between 12 and 30
→ $\Delta W(t) = \frac{R(0)}{30}$ between 16 March 2021 & 15 April 2021
→ $\Delta W(t) = \Delta R(t-180) + \epsilon \cdot \Delta V(t-180)$ for t more than 11th Sep 2021

IMPLEMENTATION SUMMARY: -

The following are the functions used in the program-
1. Constraints(prmts)
Parameters: list containing β, S(0), E(0), I(0), R(0), CIR(0)
return type: returns a list containing β, S(0), E(0), I(0), R(0), CIR(0).
This function makes sure that the parameters are within specified constraints. If parameters do not satisfy constraints, then the parameter is set to its limit.

2. PreProcessing( ):
Parameters: None
return type: pandas.core.frame.DataFrame
It returns a DataFrame containing day_num, ΔV, c_bar, Test.
This function returns a DataFrame containing t, ΔV, c⍰, T (where t is the day number starting from 16 March 2021. ΔV is the number of vaccinations per day. c_bar is running seven-day average of Δconfirmed(t). T(t) is the average number of tests done during the past 7 days. The DataFrame contains data from 16 March 2021 to 26 April 2021.

3. run_avg(arr)
Parameters: list of numbers.
return type: numpy.ndarray
This function returns a NumPy array containing the seven-day running average of the list a. When there are less than seven available days, the average of the available days is taken.

4. SEIRV (init_val)
Parameters init_val: list containing β, S(0), E(0), I(0), R(0), CIR(0)
return type: list
We create four lists S, E, I, R that contain S(t), E(t), I(t), R(t) values. For each value of t, we calculate ΔS(t), ΔE(t), ΔI(t), ΔR(t) using the equations given in problem description 1.(a), and add with the previous values before storing in S, E, I, R.
In the function, some values of S were negative. So, a condition was added so that the negative value goes to zero, and the remaining values are scaled so that their sum is 70 million. The function returns a list that contains S(t), E(t), I(t), R(t) values from t=0 to t=41. (t=0 refers to 16 March 2021, and t = 41 refers to 26 April 2021)

5. loss_func(prmts)
Parameters: list containing β, S(0), E(0), I(0), R(0), CIR(0)
return type: float
The cases-to-infections ratio is calculated for t = 0 to t = 41 using the expression CIR(t) = CIR(0)*T(t0)/T(t). where T(t) is the average number of tests done during the past 7 days and t0 is 16March 2021.Then we calculate the running seven-day average of αe(t) (where e(t) = E(t)/CIR(t)) Then we use the loss function as given in the problem description, and return the mean squared error.

6. grad(prmts)
Parameters: list containing β, S(0), E(0), I(0), R(0), CIR(0)

return type: numpy.ndarray
For estimating the gradient, β is perturbed on either side by ±0.01, CIR(0) is perturbed on either side by ±0.1, and all other parameters by ±1. It returns a NumPy array containing the gradients for each parameter.

7. grad_desnt(prmts)
Parameters: list containing β, S(0), E(0), I(0), R(0), CIR(0)
return type: numpy.ndarray
The PreProcessing() function is called to extract the values of ΔV, c⍰, T. Then the loss is calculated using loss_func(). The parameters are updated till the loss < 0.01.
Then the function returns a NumPy array containing the optimal values of β, S(0), E(0), I(0), R(0), CIR(0).

8. New_Cases_Reported()
Parameters- None
return type: list
This function returns a list of del_Confirmed values from 16 March 2021 onwards.

9. Daily_Cases_plot(result)
Parameters-result: list
result contains the predictions for S,E,I,R till 31 December 2021.
Find the average CIR value for the training period. Get E from result and reported cases by using New_Cases_Reported(). Divide E by average CIR to get the number of cases. Then plot a graph that shows the number of new cases predicted on each day till 31 December 2021. It also shows the number of new reported cases till 20 September 2021.

10. Susceptible_fraction_plot(result)
Parameters-result: list
result contains the predictions for S,E,I,R till 31 December 2021.
Get S from result. Then plot a graph that shows the evolution of the fraction of the susceptible population.

11. Open_Loop(prmts)
Parameters: list containing β, S(0), E(0), I(0), R(0), CIR(0)
return type: list
This is similar to SEIRV(). But here $\Delta V(t)$ is taken as 200000 for t >= 42. For immunity wanning we set $\Delta W(t) = \Delta R(t - 180) + \varepsilon \Delta V(t - 180)$, when t is larger than 11 September 2021. In the function, some values of S and R were negative. So, a condition was added so that the negative value goes to zero, and the remaining values are scaled so that their sum is 70 million. The function returns a list that contains S(t), E(t), I(t), R(t) values till 31 December.

12. Closed_Loop (prmts)
Parameters: list containing β, S(0), E(0), I(0), R(0), CIR(0)
return type: list
This is similar to Open_Loop(). But here the β values are updated as mentioned in the problem description. The updating starts from 27^{th} April 2021.

OBSERVATIONS AND PLOTS: -

The results after optimizing the model parameters are given below:-

```
/opt/homebrew/Caskroom/miniforge/base/envs/DA-Assign1/bin/python "/Users/dewarshinayan/PycharmProjects/DA-Assign1/DA ASn4/Assignment5.py"
Gradient Descent total iterations =  34
loss =  0.007057469495405773
Best Parameters =  [0.470504737336695, 47214999.99999966, 111999.99999546476, 272999.99998825626, 22399999.999999817, 29.549793811281727]

Process finished with exit code 0
```

The model gives **a loss of 0.007057469495405773** after 34 iterations which is less than 0.01 for the below parameters:

1. Beta = 0.470504737336695
2. S0 = 47214999.99999966 (47 million)
3. E0 = 111999.99999546476 (112 thousand)
4. I0 = 272999.99998825626 (273 thousand)
5. R0 = 22399999.999999817 (22 million)
6. CIR0 = 29.549793811281727

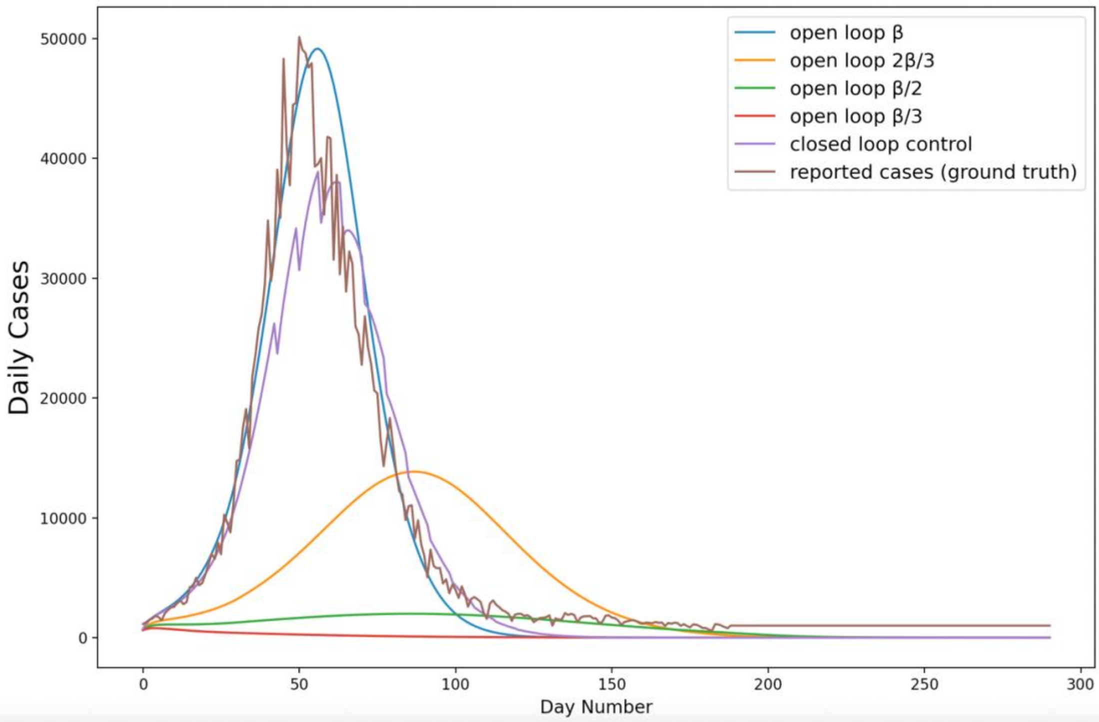

Figure 1:New Cases each day for Open and Closed Loop Methods

These parameters are used to obtain the future predictions after 19th September 2021 till 31st December 2021 and accordingly the plot is drawn.

Observations: The new daily cases are highest for BETA, followed by 2/3 BETA, 1/2 BETA and 1/3 BETA. The graph for Closed-Loop control is also shown above in Purple colour. Higher BETA values means that due to high contact rate, more people will be infected, but due to the rapid growth of the exposed, infected and recovered population, the curve also bends down very quickly because susceptible population also reduced quickly for larger BETA values. The Closed-Loop is noticeably rugged with sharp edges, and that is because of the dynamically adjusting BETA values. The ground truth is plotted in Brown colour. It shows that for some days the reported cases are lesser than the actual and for some days it's higher. For the cases of 1/2 BETA and 1/3 BETA, we can observe that the daily cases plummets down and steadily goes to zero, which means that the contact rates are too small for the disease to spread to other people quickly, and eventually the pandemic dies down because of the very low transmissibility.

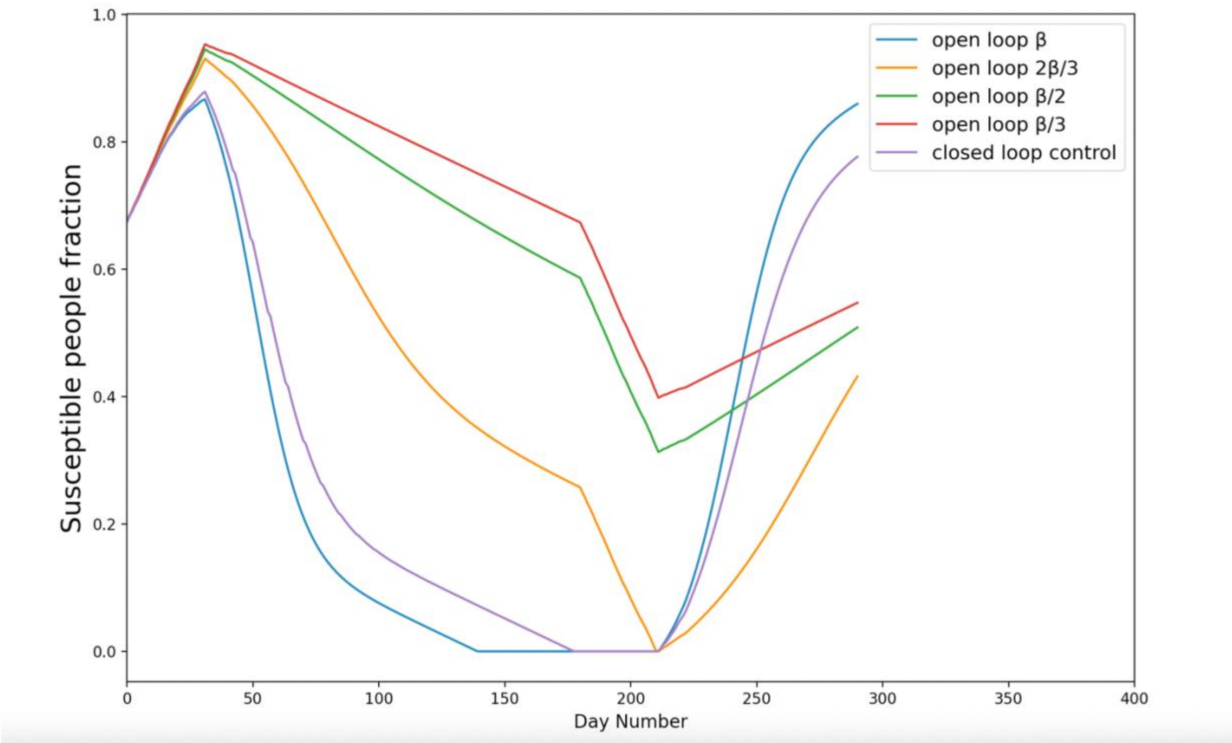

Figure 2: Fraction of Susceptible People for Open and Closed Loop Methods

Observation: Higher BETA values means that susceptible people population reduces down very quickly as higher contact rate will mean that much more people are affected and susceptibility becomes exposed very quickly. In the graph, the susceptible population reduces down more quickly for higher values of BETA (like the blue graph) as compared to lower values of BETA (like the orange, red and green graphs).