Susceptible-Infectious-Recovered Model Using MATLAB

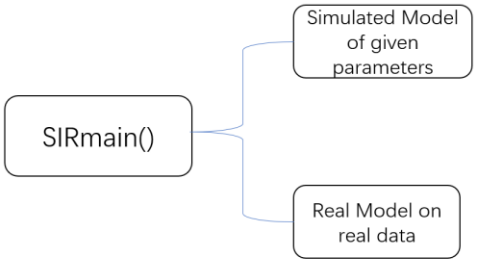
Class ID: 201930650161 Name: Ye Liqing

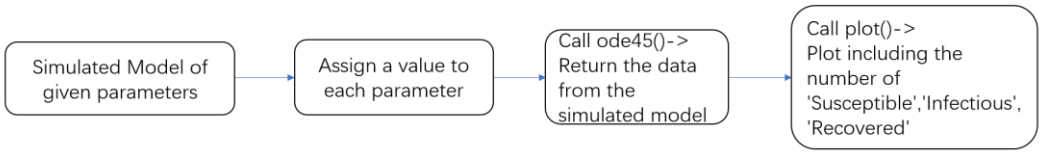
Introduction

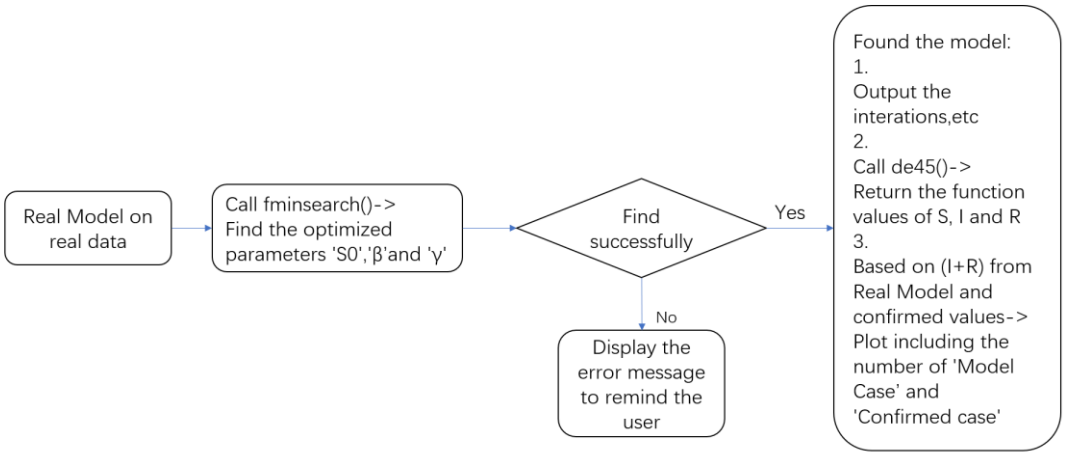
Recently, COVID-19 is rampant. A number of human beings all around the word is infected and epidemic situation is severe. Hence, an accurate Susceptible-Infectious-Recovered (SIR) is desperately needed. For optimizing the SIR model so that it can be closer to actual data, the parameters in the model must be calculated by using MATLAB. In this project, two SIR models are created. One is a simulated model of given parameters for simulating the whole COVID-19 epidemic situation including the data of susceptible populations, infectious populations and recovered populations in 200 days. Another is a real model on real data for predicting future data trends. To find minimum of unconstrained multivariable function, the project use derivative-tree method and the method of finding the square of the second norm. After obtaining the target parameters, using medium order method to solve nonstiff differential equations (*S(t), I(t) and R(t)*). Based on these data, the program plots the *graph(1)* of simulation model and the *graph(2)* of a real model in a city.

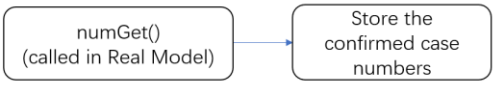
Methodology

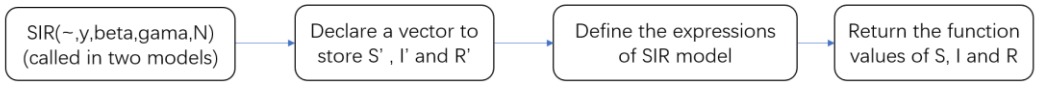
The flow charts of each function

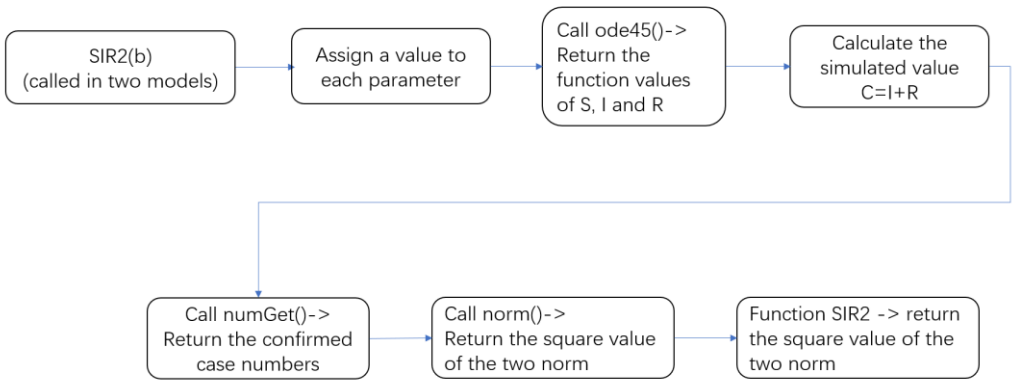





Real Model on real data
Call fminsearch()-> Find the optimized parameters 'S0','β'and 'γ'
Find successfully
Yes
No
Found the model:
1.
Output the interations,etc
2.
Call de45()-> Return the function values of S, I and R
3.
Based on (I+R) from Real Model and confirmed values-> Plot including the number of 'Model Case' and 'Confirmed case'
Display the error message to remind the user


numGet()
(called in Real Model)
Store the confirmed case numbers





Tackle the technical problems

To find minimum of unconstrained multivariable function, the project use derivative-tree method and the method of finding the square of the second norm.

Built-in function *ode45()* is a tool to find function values of ordinary differential equations using the derivative-tree method. While using it, there are some trouble arise. A syntax error occurred while assigned parameters, because of the lack of time *(t)* in expressions of *S', I' and R'*. *'~'* instead of *'t'* and correctly understand how parameters are passed can solve this syntax error problem.

Through in-depth study, the functions *norm()* can be used cooperatively to find the square value of the two norm of the difference of simulation data*(I+R)* and the confirmed case data to fit the regression equation and the confirmed data. Based on the above, the function *fminsearch()* find the best parameters *'β', 'γ'* and *'S0'* for the model.

With the help of these three functions, the target parameters are obtained.

Finally, medium order method is proved to be capable to solve nonstiff differential equations (*S(t), I(t) and R(t)*).

Based on these data, the program plots the *graph(1)* of simulation model and the *graph(2)* of a real model in a city.

Testing program

Built-in function *ode45()* has a return parameter named *exitflag* to display whether it found the accurate values *'β', 'γ'* and *'S0'*. If it had found, *exitflag* = 1, and then the program plots the results. Or it will display the message "*No accurate model was found*" to remind the users.

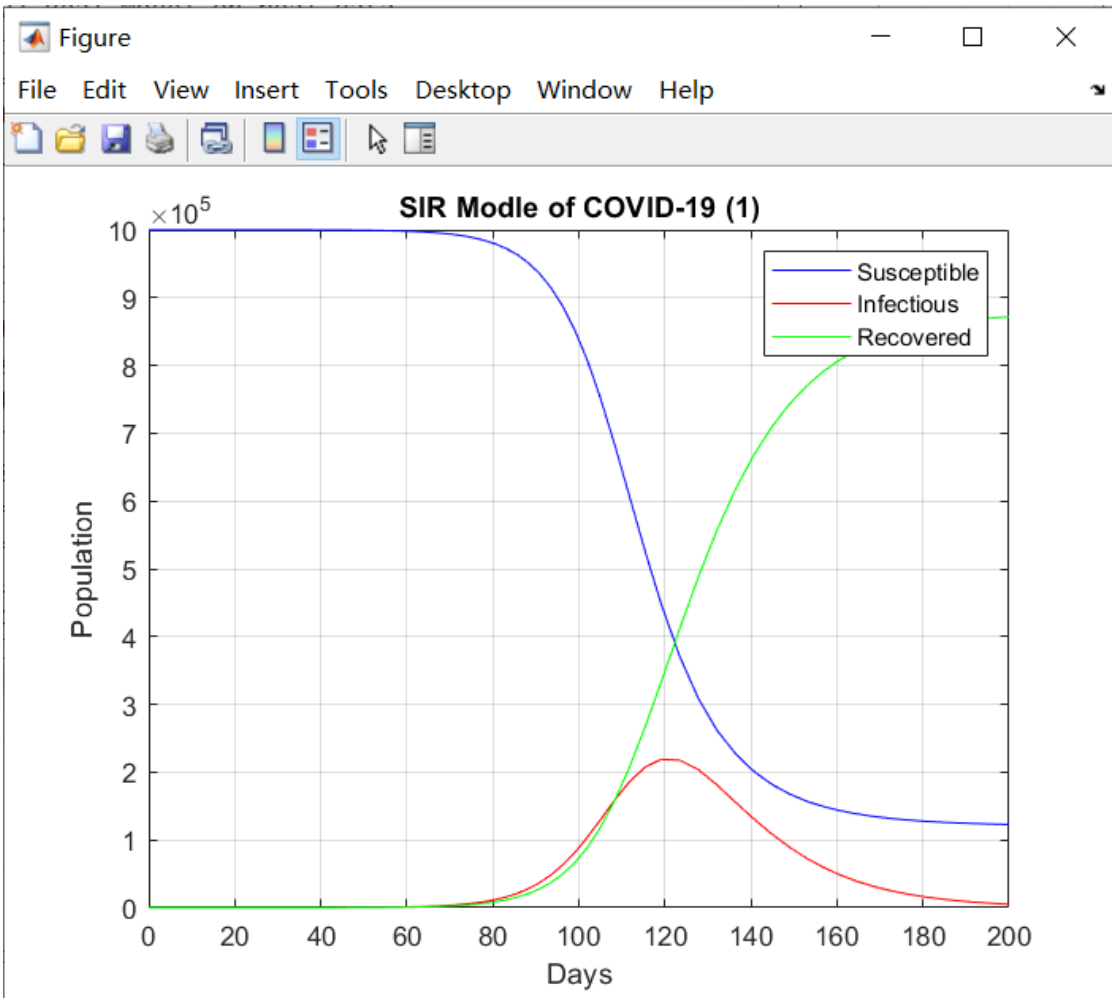
In addition, the users can also be informed of the veracity by checking the fit.

Results

Simulated Model of given parameters

This figure is a simulated model of given parameters for simulating the whole COVID-19 epidemic situation including the data of susceptible populations, infectious populations and recovered populations in 200 days.
The given parameters as follow:
β=0.2
γ=0.083
N=1000000
S0=999999
I0=1
R0=0

```
beta=0.2;
gama=0.083;
N=1000000;
ts=[0 200];
y0=[999999 1 0];
```



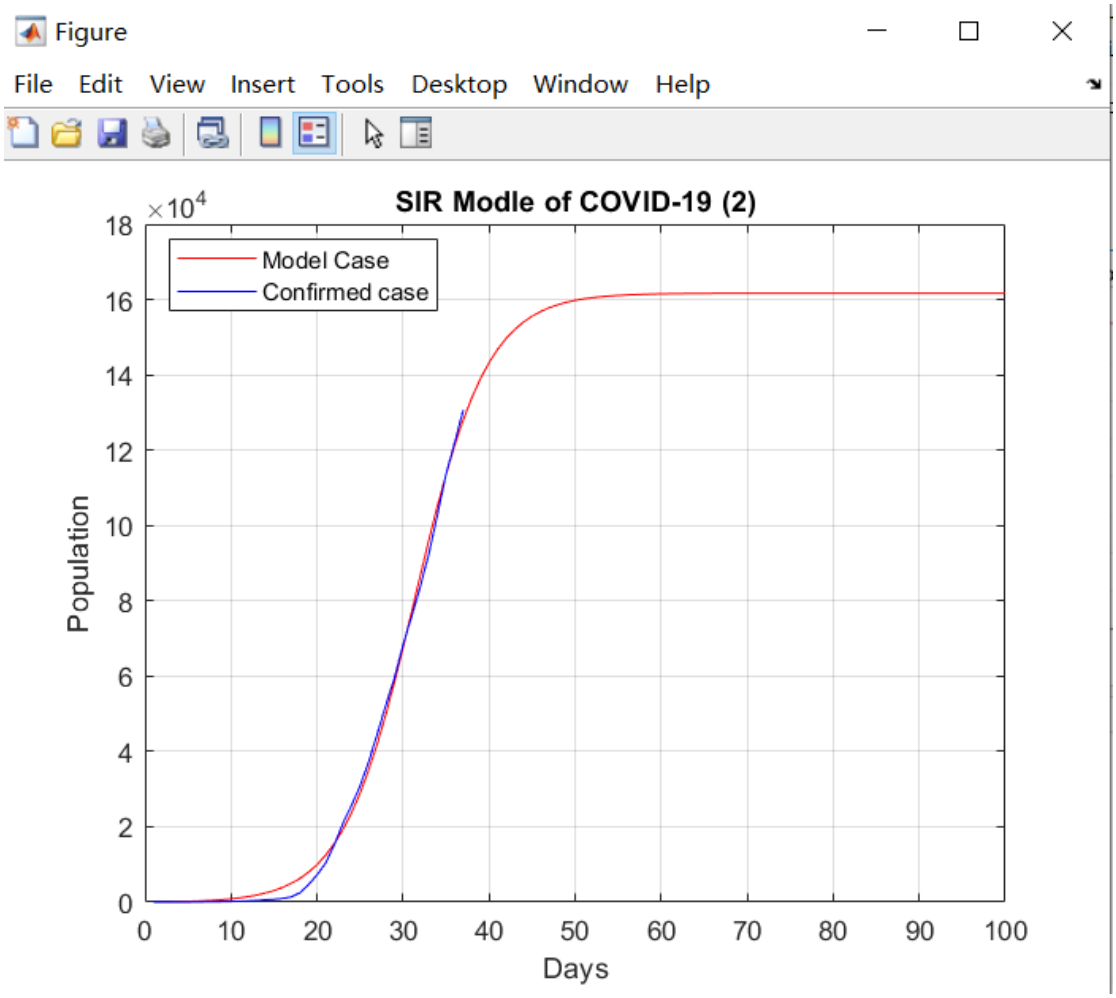
Real Model on real data

This figure is a real model on real data for predicting future data trends.
The confirmed case data in New York in 37 days as follow:

New York State Confirmed Case Number until April 6:

Date	Case Num.	Date	Case Num	Date	Case Num
2020/3/1	1	**2020/3/14**	613	**2020/3/27**	44635
2020/3/2	1	**2020/3/15**	729	**2020/3/28**	52318
2020/3/3	2	**2020/3/16**	950	**2020/3/29**	59513
2020/3/4	11	**2020/3/17**	1374	**2020/3/30**	68369
2020/3/5	22	**2020/3/18**	2481	**2020/3/31**	75795
2020/3/6	44	**2020/3/19**	4597	**2020/4/1**	83738
2020/3/7	89	**2020/3/20**	7245	**2020/4/2**	92472
2020/3/8	106	**2020/3/21**	10356	**2020/4/3**	102863
2020/3/9	142	**2020/3/22**	15168	**2020/4/4**	113704
2020/3/10	173	**2020/3/23**	20875	**2020/4/5**	122031
2020/3/11	216	**2020/3/24**	25665	**2020/4/6**	130689
2020/3/12	328	**2020/3/25**	30811		
2020/3/13	421	**2020/3/26**	37258		

This SIR model is found by the confirmed case data in New York in 37 days above. The result of predicting future data trends based on that model is in the figure as follow:



Conclusion and future development

During finishing this project, I am familiar with the use of some built-in functions in MATLAB and its syntax. When processing data, I have a deeper understanding of the SIR model. More importantly, I have precious experience about how to build a project to deal with a real problem and how to build a practical model to process data to visualize the results or predict the trends. However, I found that the iterations in function *fminsearch()* is too large sometimes when the initial parameters inputted by programmer is not close to the result. It is most waste of time and memory resources. How to find a more suitable initial parameters for the model is an area where the program deserves optimization. Nevertheless, this project can be used to fit the data of different cities and predict the trends of infectious populations. If necessary, it can also be used to found other models except SIR model by rewrite the function in *SIR.m* . Therefore, this project has extensive use, such as simulating road peak.

Appendix

User manual:

There are four files that the users are required to open in MATLAB. Three function file are named *numGet.m , SIR.m* and *SIR2.m* . The last one is a live script file named *SIRmain.mlx* .
Please store them in the same path.

After open these files in MATLAB, please run the file named *SIRmain.mlx* .

The result can be seen in the output window, including two figures fore-mentioned, and some hints.