Table of Contents


```
close all,clc,clear
load COVIDdata.mat
coviddata = [COVID_STLmetro.cases,COVID_STLmetro.deaths]; % TO SPECIFY
t = length(COVID_STLmetro.date); % TO SPECIFY

% The following line creates an 'anonymous' function that will return the cost
 (i.e., the model fitting error) given a set
% of parameters.  There are some technical reasons for setting this up in this
 way.
% Feel free to peruse the MATLAB help at
% https://www.mathworks.com/help/optim/ug/fmincon.html
% and see the sectiono on 'passing extra arguments'
% Basically, 'sirafun' is being set as the function siroutput (which you
% will be designing) but with t and coviddata specified.
sirafun= @(x)siroutput(x,t,coviddata);
```

set up rate and initial condition constraints

Set A and b to impose a parameter inequality constraint of the form A*x < b Note that this is imposed element-wise If you don't want such a constraint, keep these matrices empty.

```
A = [];
b = [];
```

set up some fixed constraints

Set Af and bf to impose a parameter constraint of the form Af*x = bf Hint: For example, the sum of the initial conditions should be constrained If you don't want such a constraint, keep these matrices empty.

```
Af = [zeros(1,3),ones(1,4)];
bf = 1;
```

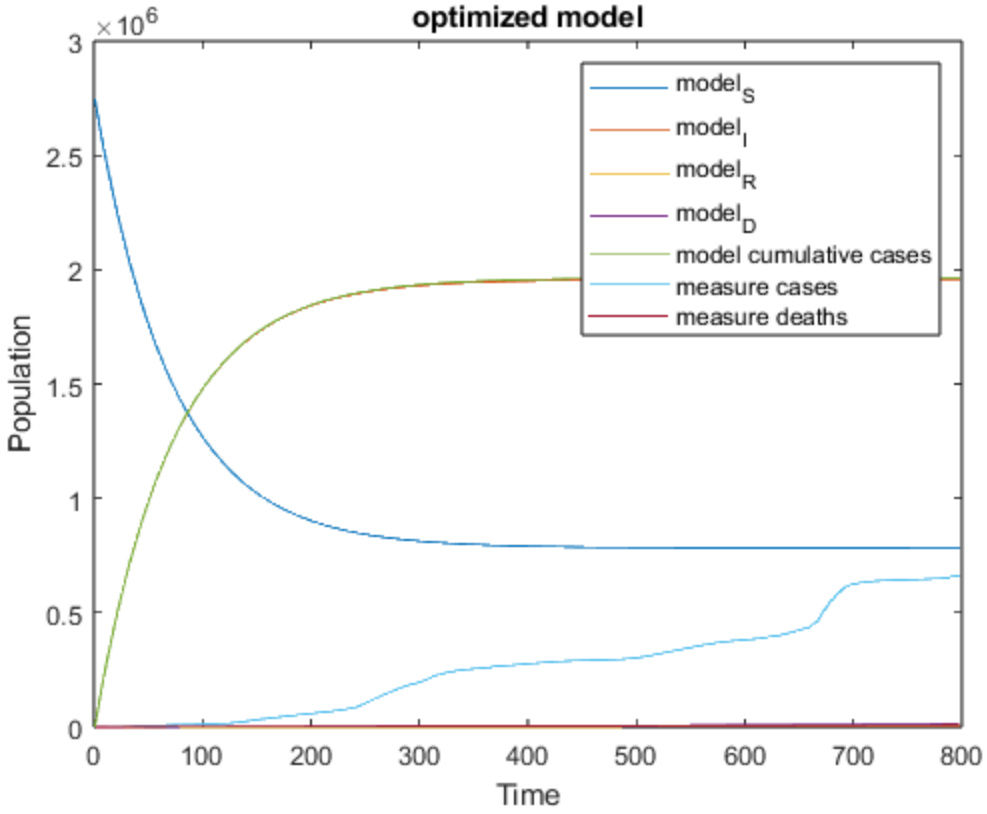
set up upper and lower bound constraints

lb < x < ub here, the inequality is imposed element-wise If you don't want such a constraint, keep these matrices empty.

```
ub = ones(1,7);
lb = [0.01,0,0,0,0,0,0];

% Specify some initial parameters for the optimizer to start from
```

```
x0 = [0.01,0.002,0.07,1,0,0,0];

% This is the key line that tries to opimize your model parameters in order to
% fit the data
[x,fval] = fmincon(sirafun,x0,A,b,Af,bf,lb,ub);
disp("optimized:")
disp(x)
disp(fval)

figure(1);
Y_fit = siroutput_full(x,t);
Y_fit = 2747143 * Y_fit;
Y_cumulative = zeros(798,1);
for i = 1:798
    Y_cumulative(i,1)=(Y_fit(i,2)+Y_fit(i,3)+Y_fit(i,4));
end
Y_fit = [Y_fit, Y_cumulative, coviddata];
plot(Y_fit);
legend('model_S','model_I','model_R','model_D','model cumulative
 cases', 'measure cases', 'measure deaths');
xlabel('Time')
ylabel('Population')
title('optimized model')
```

```
Local minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the value of the optimality tolerance,
and constraints are satisfied to within the value of the constraint tolerance.

optimized:
    0.0100    0.0000    0.0000    0.9970    0.0030    0.0000    0.0000

   1.9295e+15
```




```
% action items
t1 = 240;
coviddata1 = coviddata(1:240, :);
sirafun1= @(x)siroutput(x,t1,coviddata1);
[x1,fval] = fmincon(sirafun1,x0,A,b,Af,bf,lb,ub);
disp("t1")
disp(x1)
disp(fval)
Y_fit1 = siroutput_full(x1,t1);

t2 = 100;
Af = [
    0,0,0,1,0,0,0;
    0,0,0,0,1,0,0;
    0,0,0,0,0,1,0;
    0,0,0,0,0,0,1;
];
bf = [Y_fit1(end,1),Y_fit1(end,2),Y_fit1(end,3),Y_fit1(end,4)];
coviddata2 = coviddata(241:340, :);
sirafun2= @(x)siroutput(x,t2,coviddata2);
[x2,fval] = fmincon(sirafun2,x0,A,b,Af,bf,lb,ub);
disp("t2")
disp(x2)
disp(fval)
Y_fit2 = siroutput_full(x2,t2);
```

```
t3 = 320;
coviddata3 = coviddata(341:660, :);
sirafun3= @(x)siroutput(x,t3,coviddata3);
[x3,fval] = fmincon(sirafun3,x0,A,b,Af,bf,lb,ub);
disp("t3")
disp(x3)
disp(fval)
Y_fit3 = siroutput_full(x3,t3);
Af = [
    0,0,0,1,0,0,0;
    0,0,0,0,1,0,0;
    0,0,0,0,0,1,0;
    0,0,0,0,0,0,1;
];
bf = [Y_fit1(end,1),Y_fit1(end,2),Y_fit1(end,3),Y_fit1(end,4)];

t4 = 40;
disp("t4")
coviddata4 = coviddata(661:700, :);
sirafun4= @(x)siroutput(x,t4,coviddata4);
[x4,fval] = fmincon(sirafun4,x0,A,b,Af,bf,lb,ub);
disp(x4)
disp(fval)
Y_fit4 = siroutput_full(x4,t4);
Af = [
    0,0,0,1,0,0,0;
    0,0,0,0,1,0,0;
    0,0,0,0,0,1,0;
    0,0,0,0,0,0,1;
];
bf = [Y_fit1(end,1),Y_fit1(end,2),Y_fit1(end,3),Y_fit1(end,4)];

t5 = 98;
disp("t5")
coviddata5 = coviddata(701:798, :);
sirafun5= @(x)siroutput(x,t5,coviddata5);
[x5,fval] = fmincon(sirafun5,x0,A,b,Af,bf,lb,ub);
disp(x5)
disp(fval)
Y_fit5 = siroutput_full(x5,t5);
Af = [
    0,0,0,1,0,0,0;
    0,0,0,0,1,0,0;
    0,0,0,0,0,1,0;
    0,0,0,0,0,0,1;
];
bf = [Y_fit1(end,1),Y_fit1(end,2),Y_fit1(end,3),Y_fit1(end,4)];

Y_fit_action = [Y_fit1; Y_fit2; Y_fit3; Y_fit4; Y_fit5];
Y_fit_action = 2747143 * Y_fit_action;
Y_cumulative_action = zeros(798,1);
for i = 1:798

 Y_cumulative_action(i,1)=(Y_fit_action(i,2)+Y_fit_action(i,3)+Y_fit_action(i,4));
```

```
end
Y_fit_action = [Y_fit_action, Y_cumulative_action, coviddata];
figure(2);
plot(Y_fit_action);
legend('model_S','model_I','model_R','model_D','model cumulative
 cases', 'measure cases', 'measure deaths');
xlabel('Time')
ylabel('Population')
title('action item, distinc waves')


% action item policy
t_policy = 185;
coviddata_policy = coviddata(421:605, :);
coviddata_disired = 0.75 * coviddata_policy;
sirafun_policy= @(x)siroutput(x,t_policy,coviddata_disired);
[x_policy,fval] = fmincon(sirafun_policy,x0,A,b,Af,bf,lb,ub);
disp("policy")
disp(x_policy)
disp(fval)
Y_fit_policy = siroutput_full(x_policy,t_policy);
Y_fit_policy = 2747143 * Y_fit_policy;
Y_cumulative_policy = zeros(185,1);
for i = 1:185

 Y_cumulative_policy(i,1)=(Y_fit_policy(i,2)+Y_fit_policy(i,3)+Y_fit_policy(i,4));
end
Y_fit_policy = [Y_fit_policy, Y_cumulative_policy, coviddata_disired,
 coviddata_policy];
figure(3);
plot(Y_fit_policy);
legend('model_S','model_I','model_R','model_D','model cumulative
 cases', 'disired cases', 'disired deaths', 'measure cases', 'measure
 deaths');
xlabel('Time')
ylabel('Population')
title('COVID-19 modeled with our polcy changes')


Local minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the value of the optimality tolerance,
and constraints are satisfied to within the value of the constraint tolerance.

t1
    0.0100    0.0000    0.0000    1.0000    0.0000    0.0000    0.0000

   5.0452e+14


Local minimum possible. Constraints satisfied.

fmincon stopped because the size of the current step is less than
```

the value of the step size tolerance and constraints are
satisfied to within the value of the constraint tolerance.

t2

0.0100 0.0000 0.0000 0.3101 0.6891 0.0000 0.0008

3.0325e+14

Local minimum possible. Constraints satisfied.

fmincon stopped because the size of the current step is less than
the value of the step size tolerance and constraints are
satisfied to within the value of the constraint tolerance.

t3

0.0100 0.0000 0.0000 0.3101 0.6891 0.0000 0.0008

8.4037e+14

t4

Local minimum possible. Constraints satisfied.

fmincon stopped because the size of the current step is less than
the value of the step size tolerance and constraints are
satisfied to within the value of the constraint tolerance.

0.0100 0.0001 0.0000 0.3101 0.6891 0.0000 0.0008

7.5146e+13

t5

Local minimum possible. Constraints satisfied.

fmincon stopped because the size of the current step is less than
the value of the step size tolerance and constraints are
satisfied to within the value of the constraint tolerance.

0.0100 0.0000 0.0000 0.3101 0.6891 0.0000 0.0008

1.6430e+14

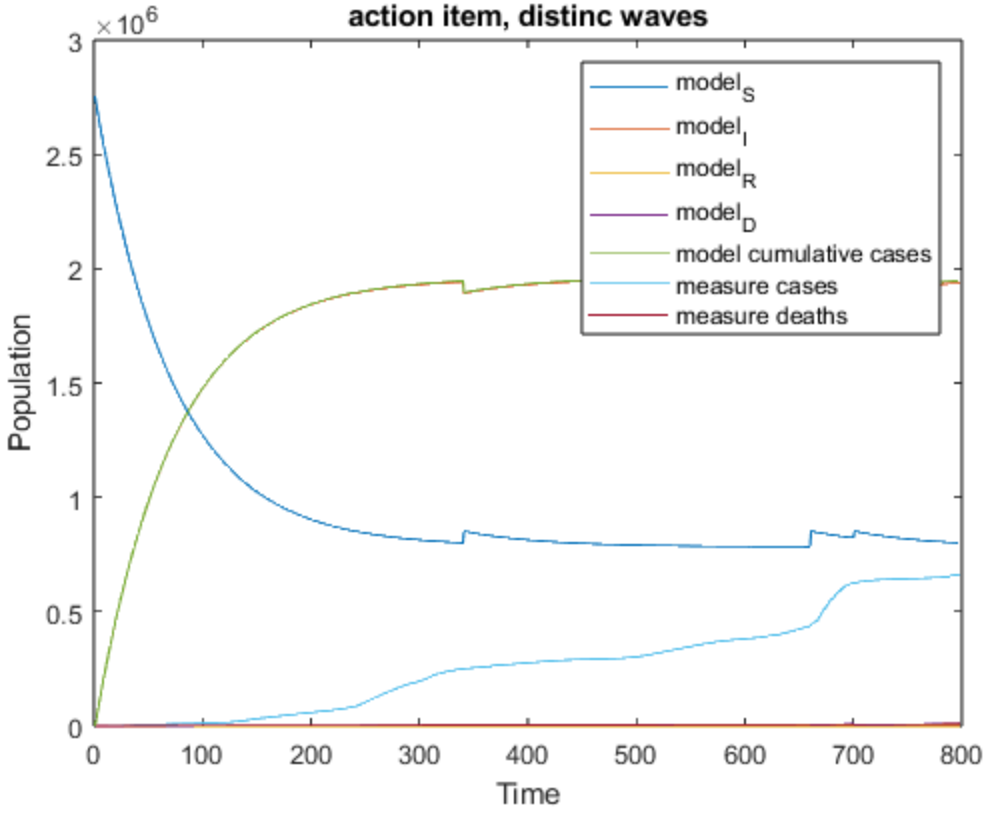
Local minimum possible. Constraints satisfied.

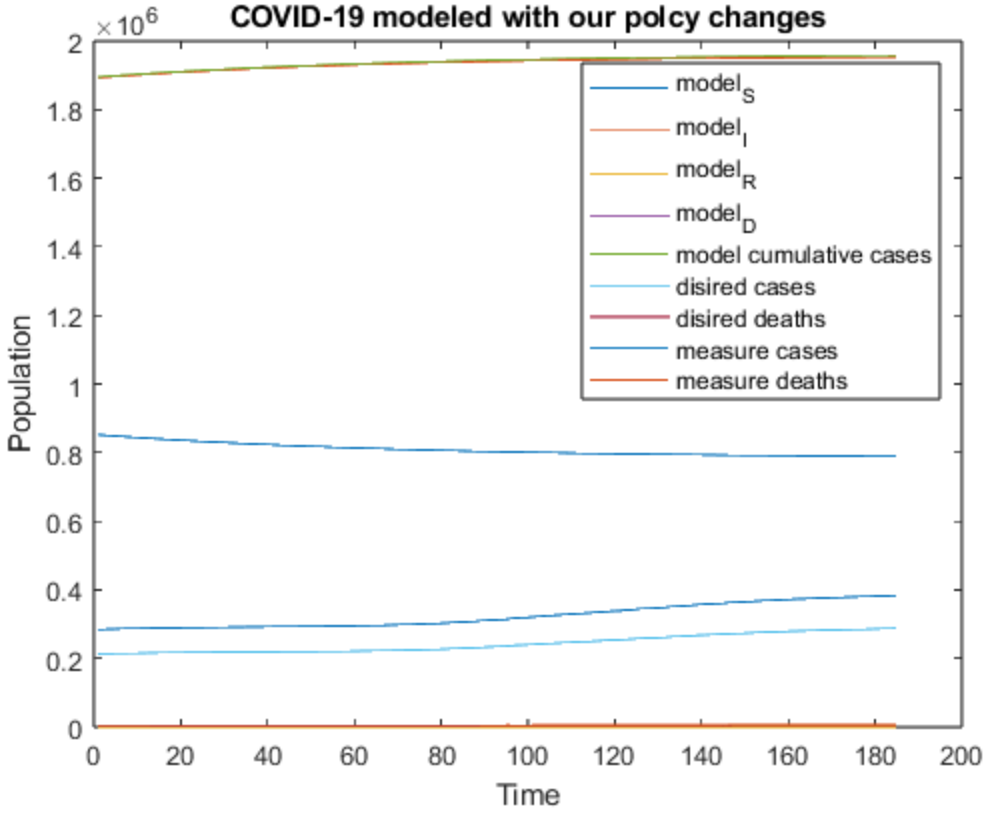
fmincon stopped because the size of the current step is less than
the value of the step size tolerance and constraints are
satisfied to within the value of the constraint tolerance.

policy

0.0100 0.0000 0.0000 0.3101 0.6891 0.0000 0.0008

5.2850e+14


action item, distinc waves
×10^6
3
2.5
2
1.5
1
0.5
0
Population
0
100
200
300
400
500
600
700
800
Time
model_S
model_I
model_R
model_D
model cumulative cases
measure cases
measure deaths



Published with MATLAB® R2022a