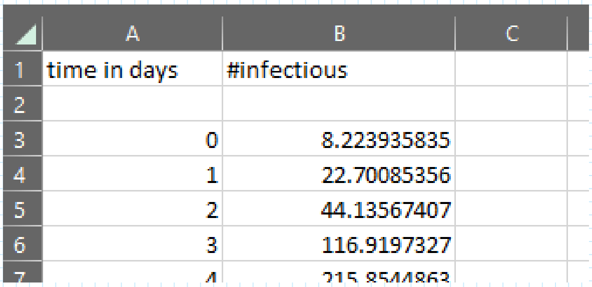
Lecture 4

24 January 2023 15:33

This last lecture on mathematical epidemiology is taken from Ch. 4 of the typed notes. It's called a "workshop" because I go through a worked example in detail. Here, I will talk through the worked example and post the relevant codes on the website.

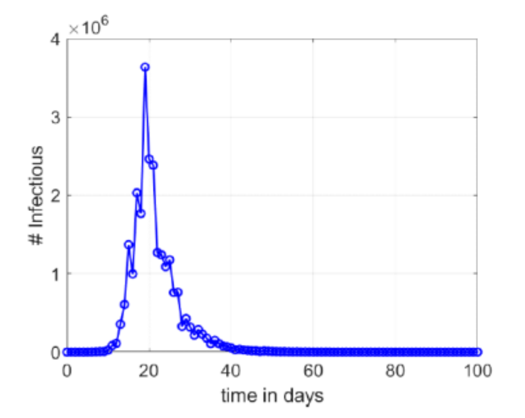
The dataset and the model (§4.1)

A sample dataset is available on the website (data1.csv)

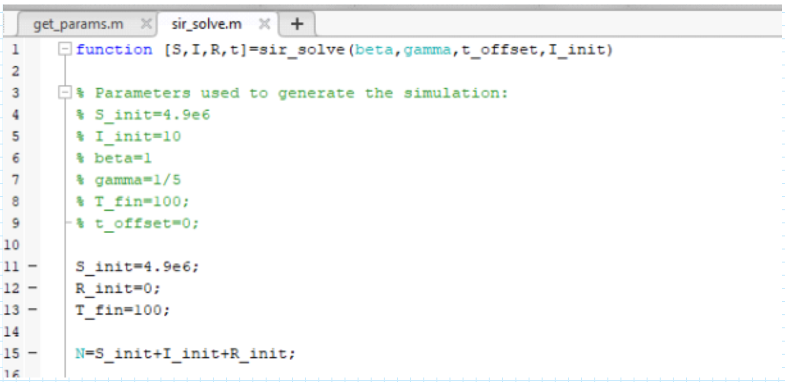


	A	B	C
1	time in days	#infectious	
2			
3	0	8.223935835	
4	1	22.70085356	
5	2	44.13567407	
6	3	116.9197327	
7	4	215.8544863	

The # infections as a $f^{\underline{n}}$ of time can be plotted:



The parameters N and T = Final Time are fixed:



```
function [S,I,R,t]=sir_solve(beta,gamma,t_offset,I_init)

% Parameters used to generate the simulation:
% S_init=4.9e6
% I_init=10
% beta=1
% gamma=1/5
% T_fin=100;
% t_offset=0;

S_init=4.9e6;
R_init=0;
T_fin=100;

N=S_init+I_init+R_init;
```

We fit an SIR model to the data:

$$\left.\begin{aligned} \frac{dS}{dt} &= -\frac{\beta}{N} SI \\ \frac{dI}{dt} &= \frac{\beta}{N} SI - \gamma I \\ \frac{dR}{dt} &= \gamma I \end{aligned}\right\} \quad t \in (-t_{offset}, T] \tag{1}$$

Thus, the simulations are started at an unknown <u>initial time</u>:

$$\text{INITAL TIME:} \qquad t = -t_{offset} < 0. \tag{2a}$$

At the initial time, patient zero is introduced to the population:

$$S(-t_{offset}) = N-1, \quad I(-t_{offset}) = 1, \quad R(-t_{offset}) = 0. \tag{2b}$$

There are three fitting parameters: $\beta, \gamma, t_{offset}$.

We propose a cost function:

$$J(\beta, \gamma, t_{offset}) = \left[\begin{array}{c} \text{Distance between model sl}^{\underline{n}} \\ \text{and data} \end{array} \right]^2 .$$

The distance is measured in the following sense:

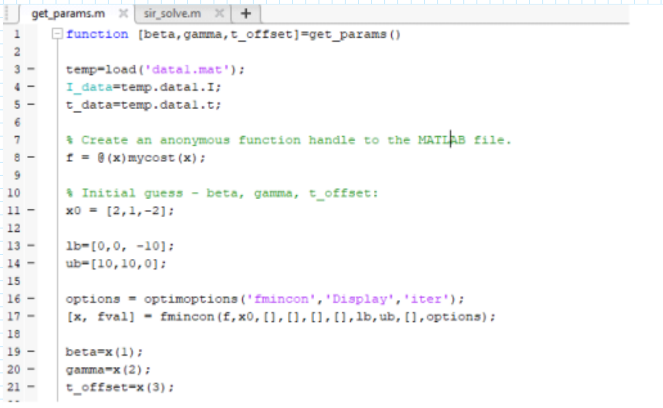
$$J(\beta, \gamma, t_{offset}) = \sum_{i=0}^{N} \left[\begin{array}{l} \log\left(I_{model}(t_i) + \epsilon\right) \\ - \log\left(I_{data}(t_i) + \epsilon\right) \end{array} \right]^2 \quad (3)$$

where:

* t_i is in days from the first day @ $t = -t_{offset}$; to the last day @ $t = T$

* ϵ is a small regularization parameter to prevent the computation of $\log(0)$ from occurring.

The cost function (3) is minimized using numerical optimization:



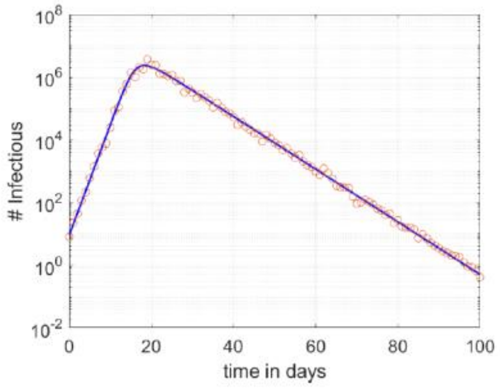
```
function [beta,gamma,t_offset]=get_params()

temp=load('data1.mat');
I_data=temp.data1.I;
t_data=temp.data1.t;

% Create an anonymous function handle to the MATLAB file.
f = @(x)mycost(x);

% Initial guess - beta, gamma, t_offset:
x0 = [2,1,-2];

lb=[0,0, -10];
ub=[10,10,0];

options = optimoptions('fmincon','Display','iter');
[x, fval] = fmincon(f,x0,[],[],[],[],lb,ub,[],options);

beta=x(1);
gamma=x(2);
t_offset=x(3);
```

The optimal parameter values are given in the table below:

Parameter	Value
β	1.0096 days^{-1}
γ	0.1999 days^{-1}
t_{offset}	$+2.7439$ days

Estimated values of β, γ, and t_{offset} for the SIR model

Graphical Results:



The SIR model fitted to the data

4.2 Error Estimation

To quantify the error associated with the estimates in Table 5.2, we perform statistical bootstrapping on the data:

- Compute the residual:

$$r_i = \log(I_{data}(t_i) + \epsilon) - \log(I_{model}(t_i) + \epsilon), \qquad i \in 1, \cdots, N.$$

- Resample the residual $\boldsymbol{r} = (r_1, \cdots, r_N)$ with replacement, to produce a resampled residual $\widetilde{r}$. From this, generate a new, 'synthetic' data set:

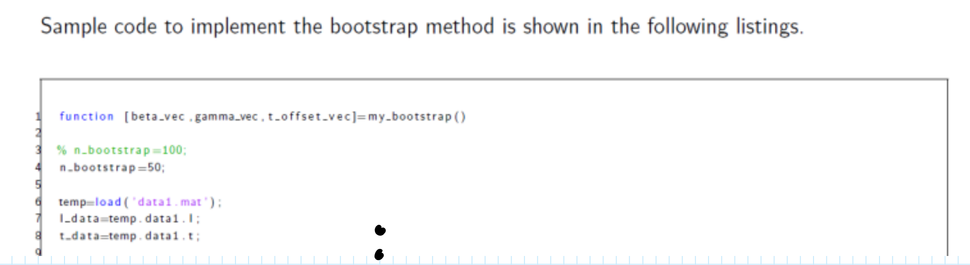
$$\widetilde{I}_{data}(t_i) = \mathrm{e}^{\log[I(t_i)+\epsilon]+\widetilde{r}_i}.$$

- Fit the SIR model to the synthetic data set, generate new estimated parameters $\widetilde{\beta}$, $\widetilde{\gamma}$, and $\widetilde{t_{offset}}$.
- Repeat, and generate a histogram of values of β, γ, and t_{offset}.

It is important to note that this is a computationally-intensive task:

- $n_{bootstrap}$ iterations of the bootstrap algorithm.
- Each iteration of the bootstrap algorithm requires a nonlinear optimization step, which requires N_{typ} ODE solves.

Thus, the typical number of ODE solves is $n_{bootstrap} \times N_{typ}$. Here, N_{typ} is not known *a priori*. It is the number of function calls required for the optimization step to converge. From running the code, this is seen to be $N_{typ} \approx 30$. For $n_{bootstrap} = 100$ this amounts to approximately $3,000$ ODE solves.

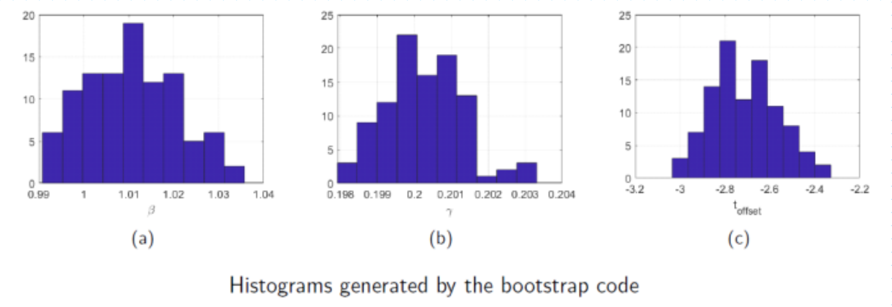


Sample code to implement the bootstrap method is shown in the following listings.

```
function [beta_vec ,gamma_vec , t_offset_vec]=my_bootstrap()

% n_bootstrap=100;
n_bootstrap=50;

temp=load('data1.mat');
I_data=temp.data1.I;
t_data=temp.data1.t;
```

⋮

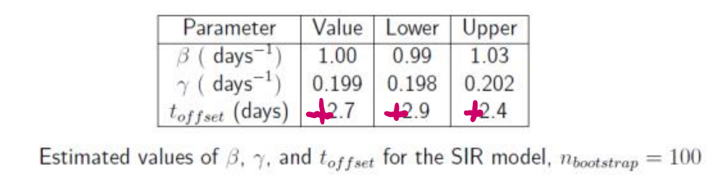
Results shown below ↴ .



(a) (b) (c)

Histograms generated by the bootstrap code

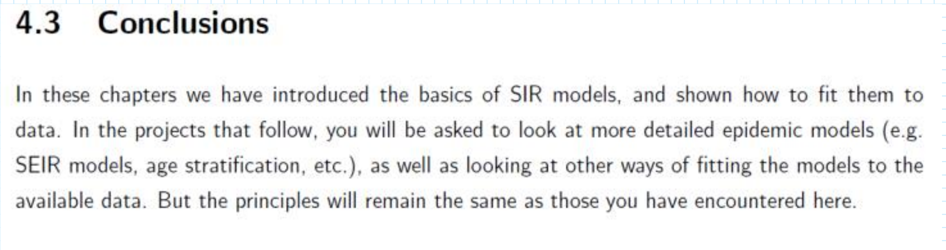
Confidence intervals are taken from the 2.5th and 97.5th percentiles of the histograms:



Parameter	Value	Lower	Upper
β (days^{-1})	1.00	0.99	1.03
γ (days^{-1})	0.199	0.198	0.202
t_{offset} (days)	+2.7	+2.9	+2.4

Estimated values of β, γ, and t_{offset} for the SIR model, $n_{bootstrap} = 100$

These results don't change if we use $n_{bootstrap}=100$ or $n_{bootstrap}=50$ meaning the results are robust.



4.3 Conclusions

In these chapters we have introduced the basics of SIR models, and shown how to fit them to data. In the projects that follow, you will be asked to look at more detailed epidemic models (e.g. SEIR models, age stratification, etc.), as well as looking at other ways of fitting the models to the available data. But the principles will remain the same as those you have encountered here.