Handout 1

July 2020

1 Basic Ideas of SIR Models

We have three populations- susceptible, infected and recovered.

$$\frac{dS}{dt} = -R_0 SI$$

$$\frac{dI}{dt} = R_0 SI - I$$

$$\frac{dR}{dt} = I$$

R_0 is called the basic reproduction number. R_0 represents the average number of people that an infected person passes the virus to early on in the epidemic.

R_0 could be constant. Other possibilities for R_0 include $R_0(S)$ or $R_0(S, I)$.

The model has been non-dimensionalized so that $S(t)$ and $I(t)$ are between 0 and 1, and the time scale is approximately in months.

It's easy to see that susceptible people are transferred to the infected population, and that infected people are transferred to the recovered population. This means that the susceptible population is a decreasing function.

One thing we can do is divide $\frac{dI}{dt}$ by $\frac{dS}{dt}$ and obtain a differential equation that is solvable for $I(S)$. There is a point where $I'(S) = 0$. This means that $I(S)$ has a maximum.

Flattening the curve means reducing the maximum number of infected people as much as possible. Large numbers of infected people overwhelm healthcare systems and cause vast economic damage as well as loss of life.

Everyone is either susceptible, infected or recovered. So if $S(t)$ is the fraction of susceptible people, and $I(t)$ is the fraction of infected people, then $S(t)+I(t)+R(t) = 1$. This means that $R(t) = 1 - S(t) - I(t)$.

What happens when the recovered population is not totally immune? It turns that mathematically, it's not possible to totally eradicate the disease but it is possible to stabilise it to a very low level. There will be multiple waves of infection.

2 Python Tools

What you will need to build the model in Python: Numpy, Scipy (scipy.integrate odeint package). All the plotting functions from MATLAB are ported into Python using the matplotib library.

You will to simulate the system for given initial conditions $S(0) = S_0$ and $I(0) = I_0$. Ideally, we want a fairly large number of susceptible people at the start and a small number of infected. We should obtain graphs of $S(t)$ and $I(t)$ and $R(t)$.

Here is an example of how to use the odeint package along with plotting. https://sam-dolan.staff.shef.ac.uk/mas212/notebooks/ODE$_{E}xample.html$

3 Intro to Lockdowns

When we have a lockdown, what we're essentially doing is changing the interactions between S and I. SI decreases because the numbers of susceptible and infected people that interact in the model decrease.

What happens if we introduce a time dependent term into the equations? Try and think about how the scenario of relaxing a lockdown might be incorporated into the SIR Model.

4 Tasks

1. Implement the SIR model above in Python with a value of R_0 between 3 and 6, and graph $S(t)$ and $I(t)$. The initial conditions should ensure that $S(0) = S_0$ is fairly high, while $I(0)$ is low (less than 1 percent).

2. Think about how you would introduce a term into these equations that incorporates the fact that recovered people are not totally immune to the disease. If you have time, implement this in Python and show us the graphs of $S(t)$ and $I(t)$.