Spr5 - Ordinary Differential Equations
S.I.R Epidemiological Model

Poynting's Python Society

3rd March 2022

Abstract

This worksheet will walk through solving the coupled ordinary differential equations that model the spread of an epidemic using the S.I.R model.

1 Introduction

The task for this week is to use the S.I.R. model from epidemiology to predict the spread of a disease. In this model, a finite population, of N individuals is categorised into three groups: *Susceptible, Infected, Recovered*. The dynamics of the numbers of individuals in each category is governed by the following coupled first order equations:

$$\frac{dS}{dt} = -\frac{\beta IS}{N}, \tag{1}$$

$$\frac{dI}{dt} = \frac{\beta IS}{N} - \gamma I, \tag{2}$$

$$\frac{dR}{dt} = \gamma I \tag{3}$$

where S, I and R represent the number of susceptible, infected and recovered individuals. β and γ are 'model parameters', which represent the rates of infection and recovery respectively.

The task will be to solve this system of three coupled first order ordinary differential equations using `scipy.integrate.odeint`[1], and plot some pretty graphs like the one shown in Fig. 1

2 The y Vector and Derivatives Function

Fortunately, this problem is already defined in terms of three first order differential equations, (rather than a single 3rd order equation) so we can directly see the vector `y` should be:

$$\vec{y} = \begin{pmatrix} S \\ I \\ R \end{pmatrix} \tag{4}$$

Now, using Eqns. 1, 2, 3, define a derivatives function, `derivatives` which calculates and returns $d\vec{y}/dt$ at a provided time, t (so the function needs to be passed a `t` variable too!!).

[1] The documentation for which can be found here.

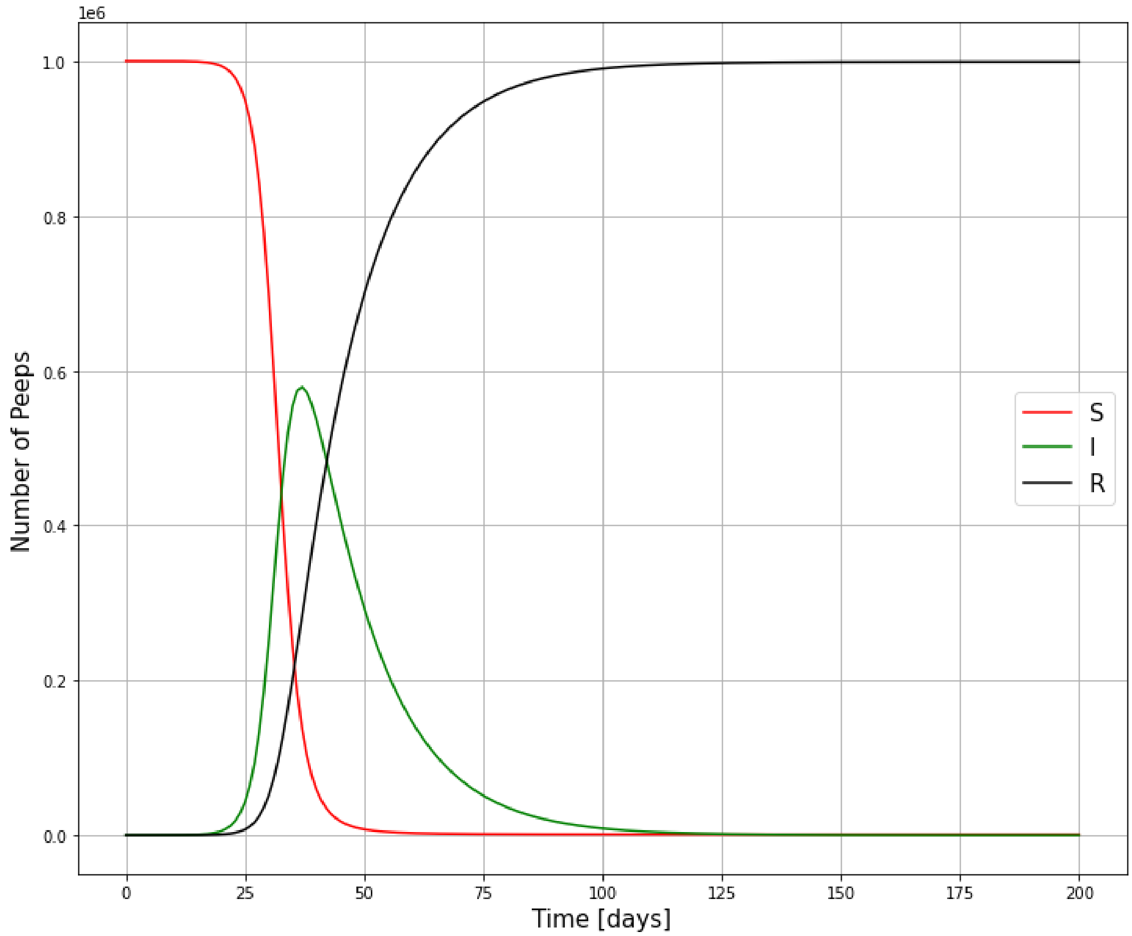

Figure 1: A single wave of infections during an epidemic, as predicted by the S.I.R. model

3 The Integration

Set up all simulation parameters, i.e. starting values for S, I, R, the model parameters and total population size, as well as a time array to integrate the ODEs over. Try starting with model parameters, $\beta = 0.5$, $\gamma = \frac{1}{14}$, but be sure to try switching out these values :)

Use `integrate.odeint` to obtain a (`len(t)` x 3) array containing the numbers of individuals in each group at each time. Be sure to pass values for β and γ as a tuple into the `args` keyword argument for the `odeint` method. This makes sures `odeint` has these parameters to pass to the `derivatives` function each time it makes a time-step.

4 Plotting

Slicing the output of the `odeint` appropriately, plot the numbers of individuals in each group over the entire range of times to check your results! How long does it take for the 'wave' to pass for different infection rates? What is the number of individuals infected at the peak of the epidemic for different model parameters?

Enjoy :)