

# CTA200H Assignment 2

Niharika Namulla

May 2021

**Question 1:** We aim to approximate the derivative of a given mathematical function 'f' at  $x_0$  using two different methods. The first approximation is derived from the Taylor series of  $f(x)$  as  $h$  approaches 0 ( $h \rightarrow 0$ ). Here  $h$  is infinitesimal.

$$d_x(f|_{x_0}) \approx \frac{f_{x_0+h} - f_{x_0}}{h} \quad (1)$$

The second and a better approximation for the derivative takes  $h$  to be finite.

$$d_x(f|_{x_0}) \approx \frac{f_{x_0+h} - f_{x_0-h}}{2h} \quad (2)$$

We can create a function for each approximation in python in the form of `def deriv(f,x0,h)` where each function approximates a derivative taking Eq(1) or Eq(2).

We then compare each numerical approximation with the analytical value (given by  $f' = \cos(x_0)$ ) to calculate and graph the error(i.e.  $\text{abs}(\text{der\_numerical} - \text{der\_analytical})/\text{d\_analytical}$ ) as a function of  $h$  on a log-log plot.

Seeing that we obtained a straight line using the log log plot we see that the size step ' $h$ ' and error in derivative hold a power law relationship.

We have  $E = kH^n$  where  $E$  is the error on the y-axis and  $H$  is the size step on the X axis such that  $\log(k)$  is a constant and ' $n$ ' is the slope.

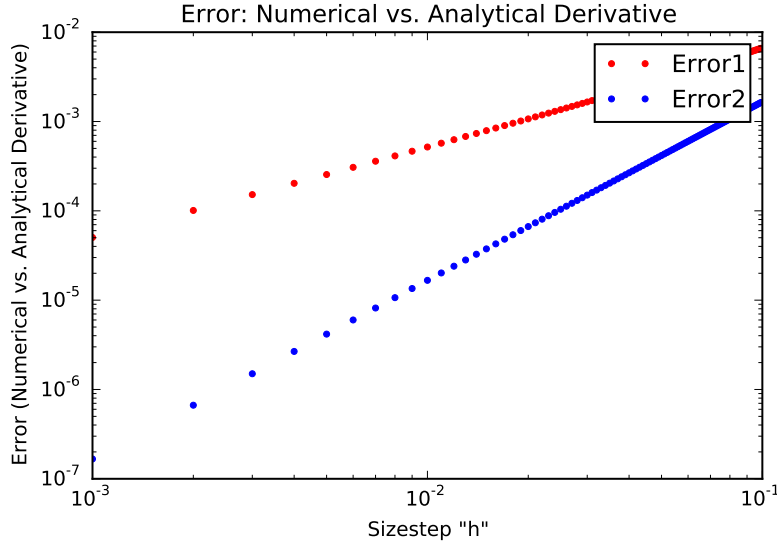


Figure 1: 'Q1'

From Figure1. we can see that the second method (blue line) is a better approximation for the derivative compared to the first (red line). The blue line is always below the red. As we increase 'h' the error in both functions is seen to increase, while the second function is seen to increasingly approach the first.

**Question 2:** We aim to iterate Eq(3) in the complex plane  $c = x + iy$  where  $-2 < x < 2$  and  $-2 < y < 2$ , starting with  $z_0 = 0$ . Doing this we see that while some points are bounded to a specific range others run off to infinity. Using various properties of Python's Matplotlib module, we are able to construct an image where values that diverge are set to be 'white' while bounded values are 'black'. Doing so we obtain a fractal like structure which can be seen in Figure 2.

$$z_{i+1} = (z_i)^2 + c \quad (3)$$

Furthermore, we are able to produce an image where the colour of each coordinate point on the complex plane varies depending on the iteration number at which the given point diverges. This can be observed in Figure 3, where a colour bar displays the iteration values and the colour hue associated with it (max iteration set to be 100).

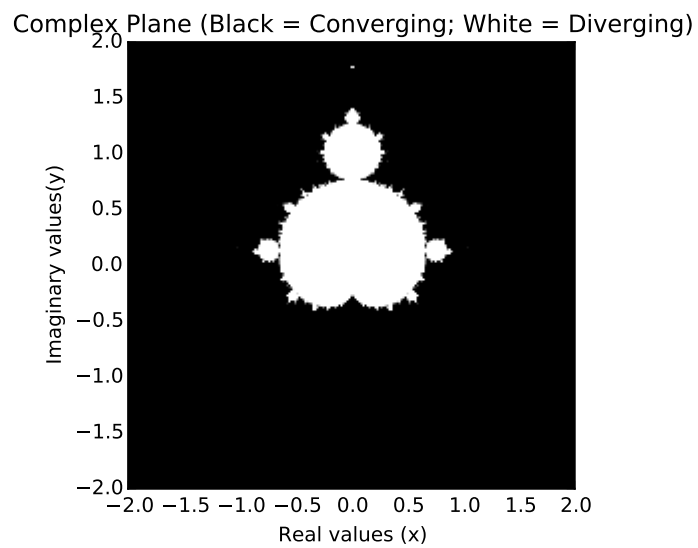


Figure 2: 'Q2A'

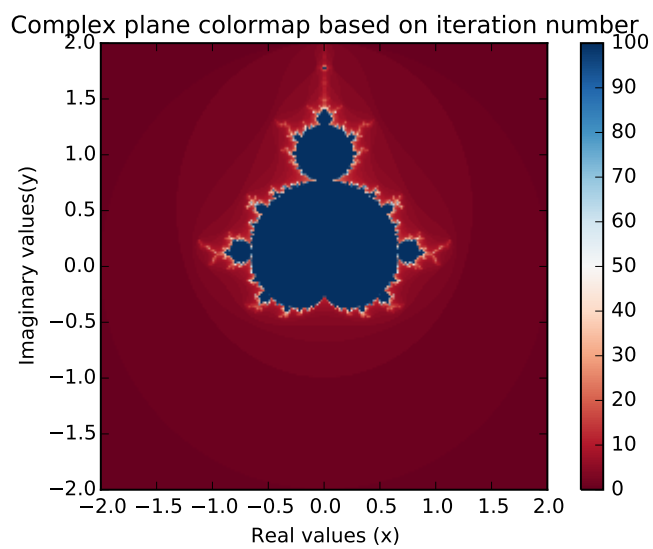


Figure 3: 'Q2B'

**Question 3:** Taking a fixed model of size  $N$  we use the SIR mathematical model of disease spread in population. Given the set of 3 first order ODE's and initial values, we are able to use 'odeint' function from scipy.integrate module to solve the ODE's and graph the resulting solutions with matplotlib module. Taking different values of Beta(transmission rate) and Gamma(recovery rate) we obtain Figure 4.

The values for Beta and Gamma must both be less than 1 as we cannot have more infected or recovering individuals than there are individuals in the population. The sum of each constant must at most be equal to 1 again since we cannot have a hundred percent of the population being infected and recovering at the same time. In all 3 scenarios we take transmission rate to be higher than recovery rate. Doing so we see the peak of the red line (Infected) shift to the left. As more and more people are infected due to higher transmission rate, it takes fewer days to reach the peak of the infected curve.

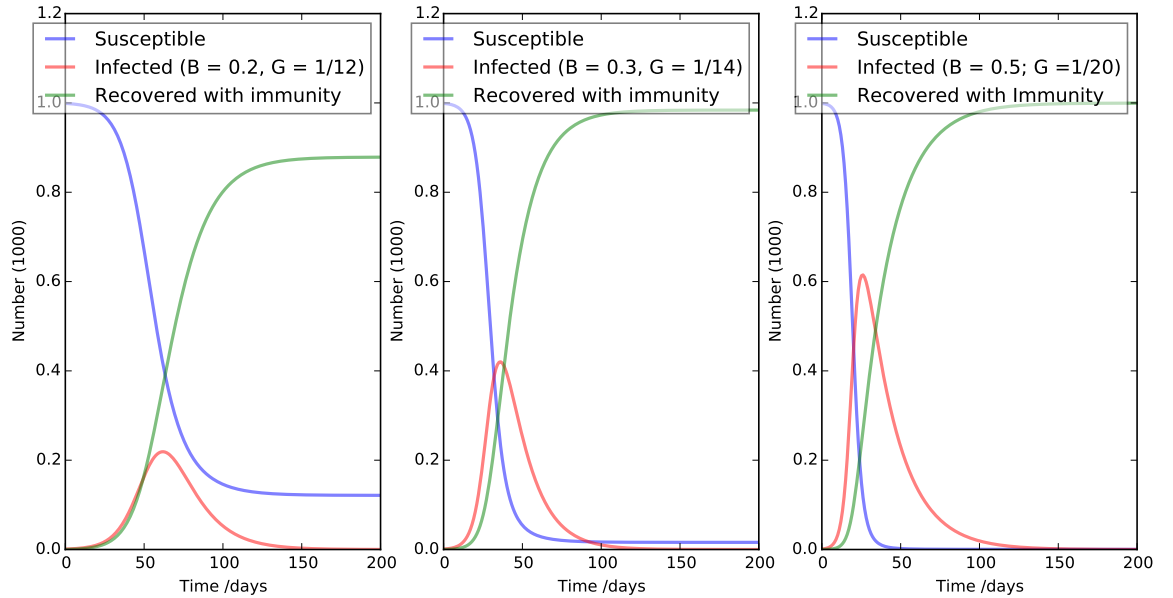


Figure 4: 'SIR Curves with varying parameters ( $B = \text{Beta}$ ;  $G = \text{Gamma}$ )'