

# CTA200 2020 Assignment 2 Summary

SURP Student Ethan Sun

May 10th, 2020

## Question 1

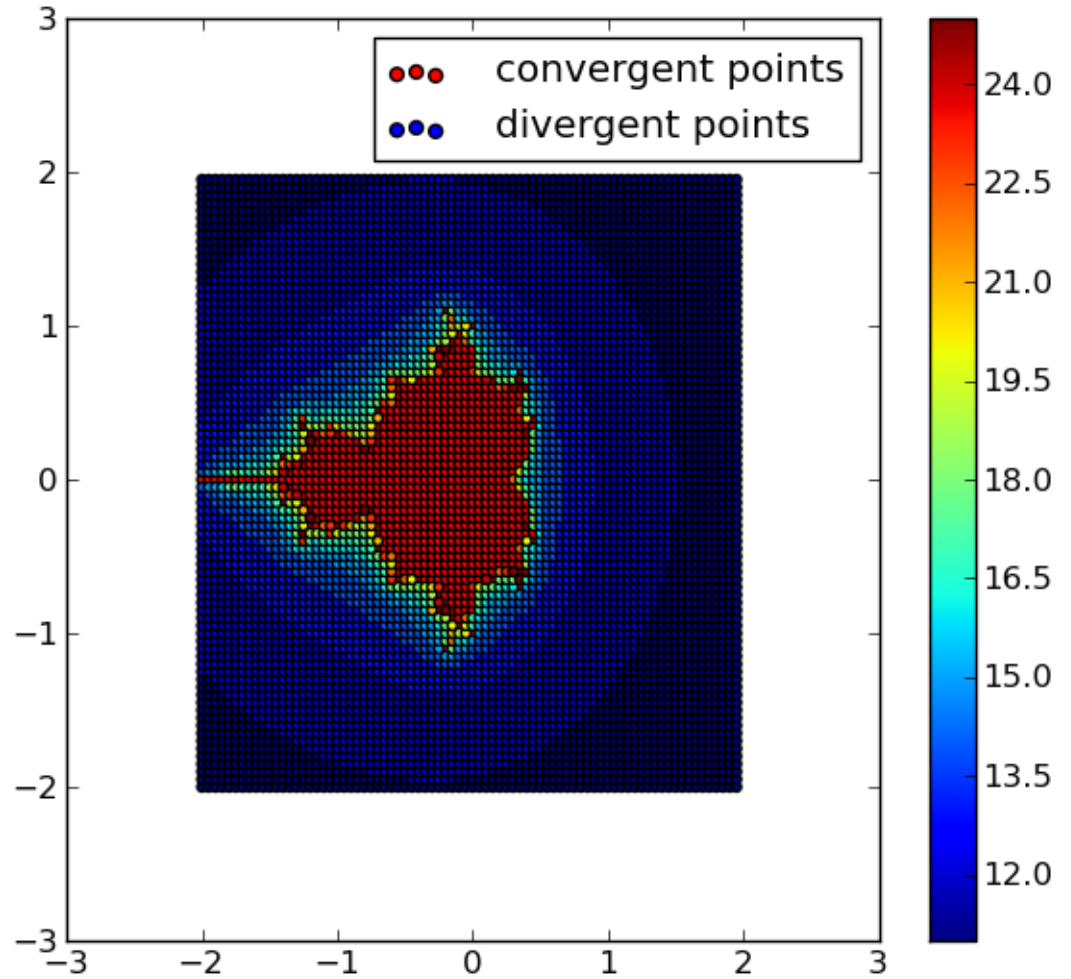
### Method

To represent the points on the imaginary plane, I used two nested "for loops", each with a numpy range "numpy.arange(-2.0, 2.0, 0.05)". I then used another for loop to calculate the  $z$  value for 25 iterations. At the end of the loops, I applied the "numpy.isfinite()" function to evaluate the real and imaginary part of  $z$ .

If both functions return "True", I would then assign the corresponding  $c$  value to a list where all the  $c$  values that result in a converging  $z$  value are stored. If either of the evaluation returns "False", I would assign the  $c$  values to a list for diverging results. In addition, I also kept track of how many iterations went through before  $z$  diverged using " $a = a + 1$ " as a counter. These counters are assigned to a list as well.

Finally, I plotted the convergent and divergent points using "plt.scatter()", and assigned the divergent iteration counter list to the "color" section in order to plot a color scale.

## Results



The convergent points are shown in red in this graph, and the rest of the divergent points are colored in gradient from blue to red. The darker blue points indicate that they diverged very early (around 11 to 12 iterations), while the brighter ones diverged after going through more iterations. It could be observed that most of the converging points have small imaginary and real components (less than 1), and as the magnitude of these two parts get larger, the  $z$  value tend to diverge after fewer iterations.

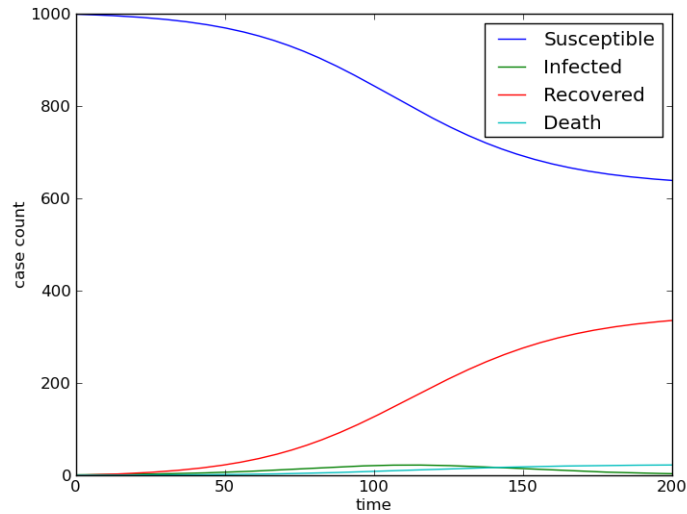
## Question 2

### Method

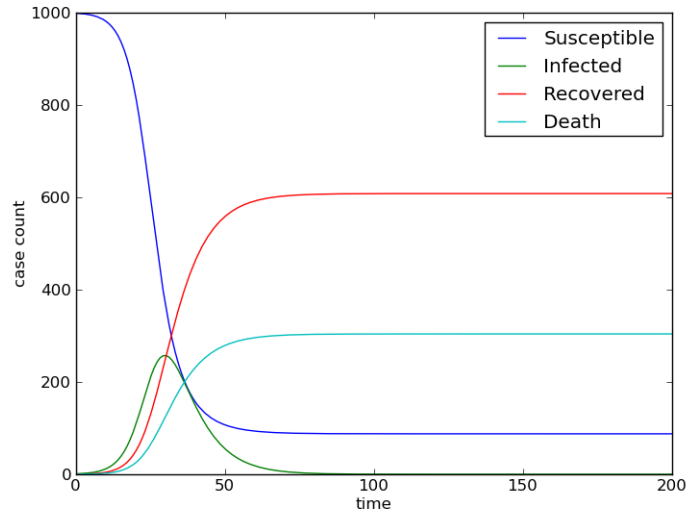
In order to provide the SIRD curves for 4 different cases, I first put all the parameters ( $\beta$ ,  $\gamma$ ,  $\mu$ ) representing infection rate, recovery rate, and mortality rate into three individual arrays. I then defined the "SIR model()" function which takes in the above parameters, time vector, as well as a variable "y" representing S, I, R, and D values. This function returns the derivative of each variable with respect to time.

To calculate the result, I created a tuple which takes the elements from the parameter arrays for each case. There are 4 cases in total, starting from a relatively moderate scenario where the infection and mortality rates are low and the recovery rate is relatively high. I then tried different combinations of parameters to simulate worse scenarios. Finally, I applied the `integrate.odeint()` function to calculate the SIRD values for each respective case before organising them into an array and plotting the results against time.

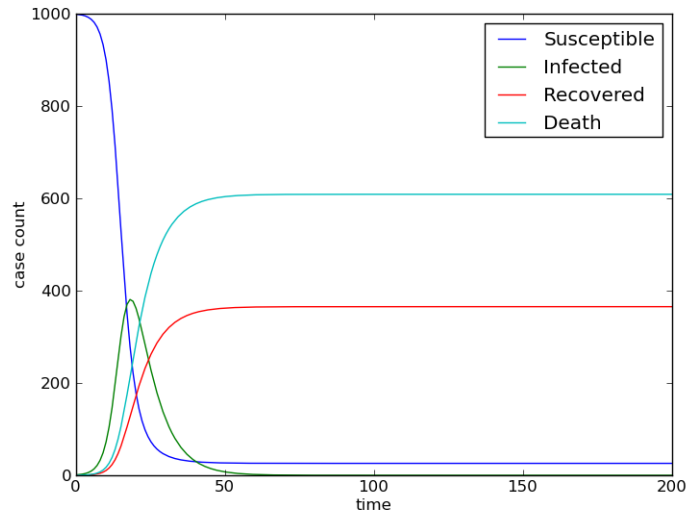
### Results



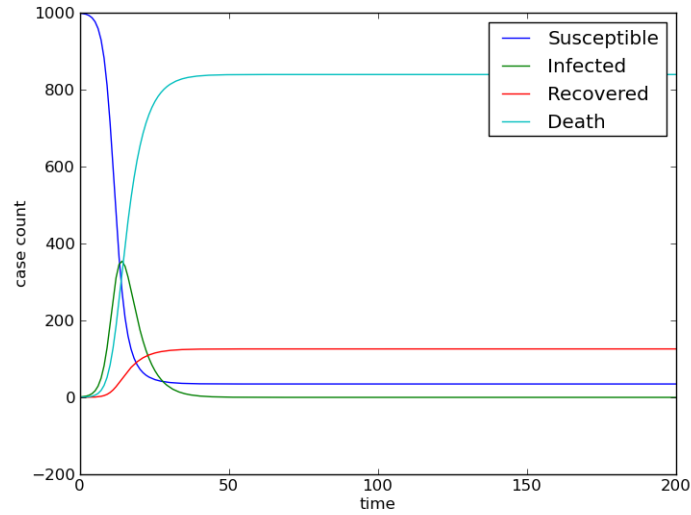
In this graph,  $\beta = 0.2$ ,  $\gamma = 0.15$ , and  $\mu = 0.01$ . Most of the infected patients recovered and the overall mortality is low. After 200 days, the spread of the disease slowed down significantly.



In this graph,  $\beta = 0.4$ ,  $\gamma = 0.1$ , and  $\mu = 0.05$ . The disease spread much faster and the number of infecte patients peaked around Day 30 . The death toll is much higher as well, but the panemic stopped after around 50 days.



In this graph,  $\beta = 0.6$ ,  $\gamma = 0.06$ , and  $\mu = 0.1$ . Similarly, the disease spread much faster and the number of infecte patients peaked around Day 20 . The death toll is, again, much higher, and the panemic stopped after around 40 days.



In this graph,  $\beta = 0.8$ ,  $\gamma = 0.03$ , and  $\mu = 0.2$ . This is the worst case scenario where the infection and mortality rates are unreasonably high, while the recovery rate is very low. The disease spread rapidly and killed almost 80% of the population in 30 days and consequently ended itself quickly.

Although the first set of parameters I tried reflected the realistic behaviour of a panemic to a certain extent, the rest of the cases are hardly reasonable or useful. For example, to reflect real life cases, it would be a better idea to use a smaller  $\beta$  value if the mortality rate,  $\mu$  is as high as 0.2, because strong quarantine measures would usually be taken swiftly by governments and healthcare services when dealing with such a dangerous disease, and the disease would be forced to become less contagious due to human interference.