Week 1

My first correspondence with Dr Graeme Auckland pertaining to the Pandemic senior honours project was on January 3rd, 2023. The following day we organised a teams meeting and discussed the goals and targets for the project. Here I assigned myself the tasks of completing the numerical integration of the SIR functions in python, reading about potential future models that may fit better, and drawing vector fields to illustrate the differential equations of pandemic modelling.

I spent the following week focusing on what the final goal of the project would be. I came to Graeme with the suggestion of trying to recreate the real UK pandemic infection curve through modelling and data analysis. This was seen as incredibly ambitious as apparently nobody has managed to achieve this before, therefore the goal was adapted to an attempt at recreating the graph and seeing how close I could get.

For this to work the SIR model would not be sufficient for COVID-19. Suffers of COVID-19 tend to have an incubation period of several days where they are symptomless, furthermore, studies suggest that people who have recovered only have temporary immunity, and vaccination efforts were implemented in two doses. Therefore, the modified version of the SEIRS model was used as this best fit the nature of the COVID-19 virus where compartments V1 and V2 were added to include vaccinated individuals with their own coefficients, Birth and death rates were then added allowing for a non-constant population. Of course, many details can be added such as age categories, COVID-19 variants, and migration. However, I consider the model to have enough detail after this. Future goals include implementing time dependant coefficients based on real world data, cleaning up the code so that it is efficient, implementing a grid-based simulation based on the models I have construct, and visualising the numerical integration in different ways to convey new information. The project is going well however I’m still not convinced of my final goal which I’m unsure I’ll be able to achieve to a satisfactory level.

Week 2

This week I further refined my Pandemic model which I’ve now called the SEIRSV2 model. I’ve also included birth and death rates in the model and drawn many graphs including a stackplot, lineplot, piechart, and vector fields. Furthermore, I’ve also created an ASCII line and grid spread of the simulations. However, the code is incredibly inefficient which is something that I plan on fixing. I’ve also become bogged down in the larger model and so I’m considering creating two models. One based on the SIRS and the other based on SEIRSV2 which much more closely relates to the covid-19 pandemic. This way, it is easier to develop, debug, and lessons applied to the SIRS model can then be applied to the SEIRSV2 model. This week I also decided to use Our World in Data as the source for all my data, in particular, the R number, vaccination rates, and infection curves can all be gathered from the same source which can be used for reference when assessing the model and be used to calculate the various coefficients.

To ensure readable code, object-oriented programming was used where the model’s coefficients would be stored as attributes and the numerically integrated functions would be calculated through methods. This also allowed for the grouping together of the ASCII models where various methods were used to simulate the grid spread. The ASCII models were constructed by creating a 2D array and updating each element of the 2D array depending on the state of its neighbours much like John Conway’s game of life.

Goals for the following week include expanding the SIRS and and graphical models, and gathering sources which should aid the model, along with preparing the first stages of the report.

Week 3

This week I studied various papers published concerning the COVID-19 pandemic to get a starting point of what the coefficients for my model should be like. I often had to manipulate the data to gain coefficients that would work for my model. For example, to get a birth-rate coefficient I took the average number of people born per day in 2020 and divided it by the population by the end of 2020. For simplicity I assumed that this would remain constant regardless of the total size of the population and that multiplying this coefficient by the current population in the simulation would give the number of births. A similar procedure was carried out for the various deathrates. Other coefficients were simply reciprocals of times, such as the recovery time being the reciprocal of 10 since the average time to recovery is roughly 10 days. Below are the sources for all my calculated coefficients.

<https://www.statista.com/statistics/281296/uk-population/>

<https://www.statista.com/statistics/281981/live-births-in-the-united-kingdom-uk/>

<https://www.ons.gov.uk/peoplepopulationandcommunity/populationandmigration/populationprojections/bulletins/nationalpopulationprojections/2018based>

<https://www.cdc.gov/media/releases/2021/p0607-mrna-reduce-risks.html>

<https://ourworldindata.org/covid-deaths?country=~GBR>

<https://www.thelancet.com/journals/ebiom/article/PIIS2352-3964(22)00584-9/fulltext>

<https://www.sciencedirect.com/science/article/pii/S2254887420301466>

<https://www.hopkinsmedicine.org/health/conditions-and-diseases/coronavirus/diagnosed-with-covid-19-what-to-expect>

<https://www.goodhousekeeping.com/health/a40062545/how-long-does-covid-symptoms-last-vaccinated/>

<https://ourworldindata.org/covid-deaths?country=~GBR>

It was also this week I made long term plans for the rest of project, listing all the major programming tasks. I placed these tasks in a to-do list app to aid in the organisation of the project.

1. Represent grid simulation graphically
2. Increase size of grid simulation
3. Draw vector fields of simulations
4. Calculate new beta values
5. Repeat previous tasks experiment with SIRS model
6. Calculate speed of spread in grid simulations

Week 4

Thanks to the temporary scaling back of the project I was able to complete some more complex tasks. I was able to graphically represent the grid simulation with matplotlib. blue represents susceptible individuals, red represents infectious individuals, and grey represents recovered individuals. Following a consistent and conventional colour scheme will minimise confusion for the reader. Rather than numerically integrating equations the total number of infections can be plotted by summing the total number of infected grid squares. However, this relies on random variables therefore it is more stochastic.

Chart, shape, square

Description automatically generatedA picture containing shape

Description automatically generated

This simplified model can still give insights into the spread of COVID-19 and it will be interesting to see how accurate it is compared to the SEIRSV2. I also drew some diagrams in PowerPoint to show how the models work without having to display the full 6 differential equations.

A picture containing spectacles, sunglasses, clipart

Description automatically generatedShape

Description automatically generated

This week I also began drafting sections of the report after completing an outline of what I want to include in the introduction with the idea being next week I would have at least 5 pages I could choose from to show my supervisor. Beyond this I did some basic cleaning of the code to improve efficiency, implemented a single method to generate vector fields rather than 3, and introduced time dependant coefficients in the model by implementing real world data. In the following week I plan to tune my coefficients so that I can more closely represent what the real infection curve looks like. Furthermore, I plan to use data from deaths to work around the limited testing in the early days of the pandemic.

Week 5

