**Methodology**

We decided to use Python programming language with the NumPy and matplotlib libraries. Python is a common program language with plenty of prior research. Its suitable at visualizing data.

In the project, we aim to create different simulations:

* A simulation for the infected, susceptible, and recovered groups.
* Perform logistic fitting on the cumulative infected population.
* Measure the error from the sample simulation.
* Introduce random noise to the simulation in order to add the external factors into consideration.

The simulations will use the SIR Model: splitting the population into 3 groups (the susceptible, infected, and recovered). There are 2 transitions in the model: a susceptible individual becoming infected and an infected individual recovering.

The simulation shows how the different population groups change over time. Initially, there were an exponential increase in the number of cases. The infected population changes reflect an epidemic wave: the infected population will reach its maximum; the epidemic reaches its peak. As the infected population grows, the recovery rate increases, and more people will recover than infected so the infected population will decrease and eventually the epidemic will end.

**Pseudocode**

**A simulation for the infected, susceptible, and recovered groups.**

The simulation is a CTMC SIR Model with the initial parameters:

1. , and constants: infection and recovery constant
2. Initial sample population: 1000
3. Simulation time: 30 days

Pseudocode

* Initialized the different population groups: the infected, susceptible, and recovered groups. Initially 10 people are simulated as infected, the rest of the population are susceptible to the epidemic, and 0 recovered individuals.
* While there are infected patients, and the time is within the limit
* Calculate the infection constant a and b
* The infection constants are used to calculate whether a patient has recovered or whether there are newly infected individuals.
* If the randomly generated probability is less than a value, then simulate a new infection otherwise simulate a new recovery
* Update time through the formula:
* Plot the times with the infected population

Chart

Description automatically generated

**A simulation to generate prediction curve alongside the simulation data.**

An additional logistic fitting function is applied to fit the simulated data using the formula:

The logistic fit function is applied on the infected data to help with the prediction of the future infection cases, the simulation data can be helpful at giving an insight for the prediction of the future real-life outbreaks.

**A simulation to apply fitting to different dates.**

The simulation considers different dates as stages in the pandemic. Using each date to simulate the projections after the date helps to give us better insights about the pandemic data at each stage in the pandemic. It is also easier to record the data in this way because it is a more realistic method of simulation: data is collected at specific time intervals – in real-life, it’s not possible to continuously update the data. This data collection is more realistic at simulating the pandemic cases.

1. Last day of the simulation, projection:
2. Average it over 100 simulations.
3. Produce a figure for the simulation

**A simulation to calculate the average error for measuring the infected patients.**

1. A simulation generates the synthetic data to model the real-life data.
2. Logistic fit is performed every 3 days on the synthetic data to generate simulated data.
3. Compare the last day data of the simulated data with the synthetic data (real-life) to calculate the error in the measurement of the infected patients. The last day is considered because the data is cumulative so the previous patients are considered.

**Result**

Chart

Description automatically generated

Figure 1. Epidemic Graph showing the infected, recovered, and susceptible populations over a specific period of time

Chart

Description automatically generated

Figure 2. Comparison between the cumulative infected data with the logistic fit data

Chart, line chart

Description automatically generated

Figure 3. Projection of the number of cases at regular date interval

Chart

Description automatically generated

Figure 4. Epidemic simulation including the noise data

Problems related to the data collection ✅

Scenario: perfect data and applying without noise ✅

Complicate the model ✅

Description: pseudocode for the program.

Graph 1: SIR graph ✅ 🗎

Graph 2 : cumulative infection data and fit ✅ 🗎

Logistic fit uses all data ✅ 🗎

Apply fittings to different dates✅ 🗎

Graph 3 : fitting from different dates. ✅

Add noise to the data points. ✅ - 3

Consider data points in the networks. ❎ - 4

Produce an average error plot. ❎ - 1

Compare the error with the epidemic trends. ❎ - 2

🗎 : written documentation for the technical solutions.

Another approach to plot the curve involves using the fitting curve from the start to the end of simulation. The simulation splits the data into different stages, each stage acts as a check point to simulate the data after. At the point, the future projection is applied to simulate the current unknown data and projection is compared with the simulated data to determine the error in the curves.