PART 1

Problem Statement

The Ministry of Health is tracking an infectious disease that is spreading through the population. It was initially thought that individuals who were infected and then recovered were immune to reinfection. As such, the Ministry was using the SIR model to track and predict the spread of the disease. However, recent data suggests that individuals who are infected and then recover can be reinfected. The Ministry has hired you, as a consultant, to modify the SIR model to a SIS model.

Introduction

The Ministry of Health is currently navigating the intricate landscape of an ongoing infectious disease outbreak that poses a critical public health concern. The situation is dynamic and managing it requires accurate and effective models for predicting disease spread. Initially, the Ministry relied on well-established Susceptible-Infected-Recovered (SIR) model. This model was predicated on the long-standing standing medical assumption that individuals who recover from the infection develop immunity, thereby moving from the "Infected" state to the "Recovered" state without the possibility of returning to the "Susceptible" state.

However, recent clinical studies and patient data have thrown a wrench into this assumption. New evidence suggests that recovered individuals can, in fact, be susceptible to reinfection. This revelation has far-reaching implications for disease control strategies, and renders the existing SIR model inadequate for the current crisis.

Acknowledging the urgency and complexity of the situation. The Ministry has engaged us as a consultant to undertake a significant adaptation of the existing model. Specifically, we have been tasked with transforming the SIR model into a Susceptible-Infected-Susceptible model, which accounts for the potential of reinfection.

Given the evolving nature of the infectious disease and the new evidence suggesting the possibility of reinfection, numerical methods and simulations offer a robust and flexible approach for addressing this complex issue. The SIS model, which needs to account for the dynamic possibility of reinfection, inherently involves non-linear equations that describe the transitions between the "Susceptible" and "Infected" states. Numerical methods are particularly apt for solving such complex, non-linear equations, allowing for a more accurate representation of real-world scenarios.

Moreover, simulations provide the capability to test various scenarios and parameters in a controlled, virtual environment. This is crucial for the Ministry of Health, as it allows for the exploration of various 'what-if' scenarios, such as changes in infection rates or the introduction of new public health measures. With simulations, it becomes possible to assess the impact of different variables on the disease's spread, something that is essential for a disease where immunity is not guaranteed post-recovery.

Therefore, the combination of numerical methods for solving the complex equations of the SIS model, along with simulations for scenario testing, offers a comprehensive toolkit. This methodology is not only aligned with the immediate need for an adaptable model but also offers the flexibility to incorporate new data and assumptions as the situation evolves. This is invaluable for the Ministry's ongoing efforts to control the outbreak effectively. This comprehensive report outlines the rigorous methodology employed in adapting the model, including the mathematical models and computational techniques used. It also presents the new SIS model, detailing how it differs from its SIR counterpart and what these differences mean. Finally, concluding with an evaluation of the model's effectiveness and offer recommendations for its practical application in the ongoing efforts to control the outbreak.

Methodology

In response to the evolving understanding of the infectious disease under study, a transition was made from the Susceptible-Infected-Recovered (SIR) model to the Susceptible-Infected-Susceptible (SIS) model. The key distinction between these models lies in their interpretation of recovery: the SIR model assumes post-recovery immunity, whereas the SIS model accounts for the potential of reinfection after recovery.

Model Modification

To simulate the disease spread, the Runge–Kutta method was implemented. The simulation considered a population of one million individuals and initiated the transmission with a single infected individual, often referred to as "patient zero."

The following variables were considered:

- S - Proportion of the susceptible population.
- I - Proportion of the infected population.
- β - Infectious contact rate.
- γ - Recovery rate.

The model was built on the following assumptions:

1. The probability that a susceptible individual becomes infected upon contact with an infected individual is 50%.

2. An infected individual remains sick for a duration of 100 days.

Mathematical Representation of the SIS Model

The SIS model is formulated using a set of ordinary differential equations (ODEs) as follows:

$$\frac{dS}{dt} = -\beta SI + \gamma I,$$
$$\frac{dI}{dt} = \beta SI - \gamma I.$$

Normalization

Normalizing the population to 1 simplifies the interpretation of the model parameters such as β and γ. This normalization facilitates comparisons across different population sizes and demographics.

Derivation of the Fourth-Order Runge-Kutta Method

Mathematical Formulation

Consider a generic first-order ODE:

$$\frac{dy}{dt} = f(t, y).$$

The aim is to approximate $y(t)$ for t in the interval $[t_0, T]$, given an initial condition $y(t_0) = y_0$.

Core Principles

The essence of Runge-Kutta methods lies in approximating the function $y(t)$ by incrementing its current value through a weighted sum of several estimators. These estimators are derivatives calculated at various points within the interval $[t_n, t_{n+1}]$, where $t_{n+1} = t_n + h$ and h is the step size.

Full Derivation

For the fourth-order Runge-Kutta method, the k_i terms for S and I are derived as follows:

1. The k_i terms for S are derived as follows:

$$\begin{aligned}
k_{1,S} &= \Delta t\left(-\beta SI + \gamma I\right), \\
k_{2,S} &= \Delta t\left(-\beta(S + \frac{k_{1,S}}{2})(I + \frac{k_{1,I}}{2}) + \gamma(I + \frac{k_{1,I}}{2})\right), \\
k_{3,S} &= \Delta t\left(-\beta(S + \frac{k_{2,S}}{2})(I + \frac{k_{2,I}}{2}) + \gamma(I + \frac{k_{2,I}}{2})\right), \\
k_{4,S} &= \Delta t\left(-\beta(S + k_{3,S})(I + k_{3,I}) + \gamma(I + k_{3,I})\right).
\end{aligned}$$

2. Similar terms can be derived for $k_{1,I}, k_{2,I}, k_{3,I}, k_{4,I}$.

3. The new values of S and I are calculated as:

$$\begin{aligned}
S_{\text{new}} &= S + \frac{1}{6}(k_{1,S} + 2k_{2,S} + 2k_{3,S} + k_{4,S}), \\
I_{\text{new}} &= I + \frac{1}{6}(k_{1,I} + 2k_{2,I} + 2k_{3,I} + k_{4,I}).
\end{aligned}$$

Implementation of ODEs in Numerical Simulations

The ODEs central to the SIS model were implemented using the Runge-Kutta 4th order method to ensure high accuracy. Model parameters and initial conditions were carefully chosen based on empirical data and scientific literature. The model was also calibrated by comparing simulation outcomes with real-world epidemiological data.

Runge-Kutta Implementation

A loop was set up to iterate through time steps, starting from an initial time t_0 up to a final time T. At each time step, k_1, k_2, k_3, and k_4 were calculated according to the derived equations. These values were then used to update S and I for the next time step.

Model Validation

The final phase of the methodology centers on model validation, which involves comparing numerical solutions to analytical solutions. A high degree of alignment between the numerical and analytical solutions serves as a strong validation of the model.

Application of the Runge-Kutta Method in Python

The Runge-Kutta 4th order method was implemented in Python using a loop structure to iterate over a predefined time range. The function signature is as follows:

```
def runge_kutta_sis(S0, I0, beta, gamma, dt, T):
    ...
```

This implementation enables simulation of the SIS model under various conditions, providing valuable insights for effective disease control and resource allocation.

Sensitivity Analysis

In the study, sensitivity analysis is crucial for validating the robustness and reliability of the outcomes derived from the SIS model, formulated as a set of ordinary differential equations (ODEs). The approach involves slightly modifying the initial conditions and key parameters of the ODEs within a biologically plausible range to assess their impact on the system's dynamics, specifically the rates of infection and recovery over time. Furthermore, to extend the analysis to scenarios where multiple parameters are altered simultaneously, thereby offering a more comprehensive view of the system's sensitivity to a combination of uncertainties. This multi-parameter sensitivity analysis allows identification of critical variables that have a significant influence on the system's behavior and could, therefore, be targeted for more effective public health interventions and resource allocation. The insights gained from the sensitivity analysis are invaluable for understanding the model's vulnerability to changes in key variables and conditions, and they play a pivotal role in assessing the model's applicability to real-world epidemiological scenarios.

Results

Investigate how the spread of the disease changes as β varies from 0.1 to 1

For this investigation, we kept the step size at 1 day and varied β from 0.1 to 1. The values used were $\beta = 0.1, 0.3, 0.5, 0.7, 1$.

The relationship between β values and the dynamics of disease spread is evident in the plotted results. Higher β values correspond to a notably elevated peak of infection, accompanied by a swifter propagation of the disease throughout the population. This observation underscores the pivotal role that the infectious contact rate (β) plays in shaping the speed and intensity of disease transmission. The increase in β leads to a more rapid accumulation of infected individuals, highlighting the urgency of controlling and managing the infectious contact rate as a critical factor in mitigating disease outbreaks.

Investigate how the spread of the disease changes as γ varies from 0.01 to 0.1

For this part, the step size was kept at 1 day and γ was varied from 0.01 to 0.1. The values used were $\gamma = 0.01, 0.03, 0.05, 0.07, 0.1$.

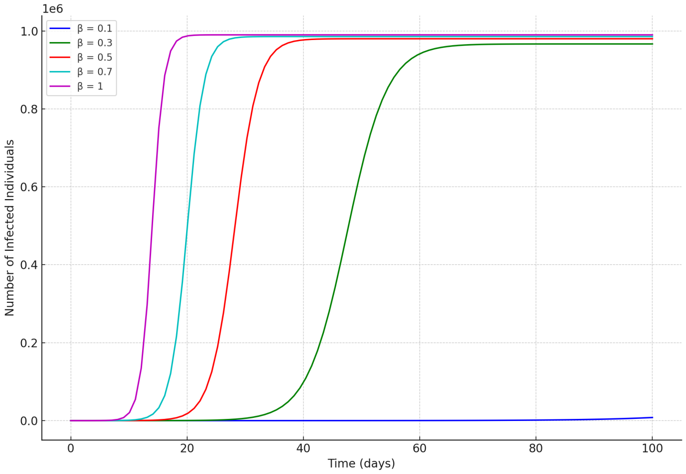

Figure 1: Effect of varying β on the SIS model (Runge-Kutta 4th-Order Method)

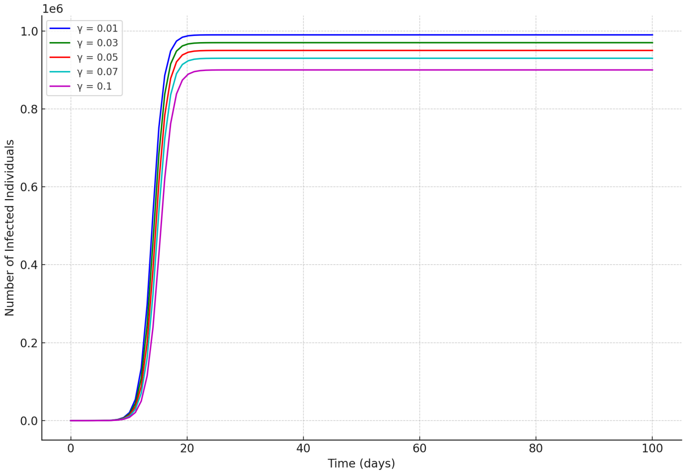

Figure 2: Effect of varying γ on the SIS model

An analysis of the simulations reveals a relationship between the recovery rate γ and the dynamics of the disease spread. Specifically, higher values of γ are associated with a more rapid decline in the number of infected individuals, leading to a lower peak in the infection curve. This suggests that an increase in γ effectively reduces the overall burden of the disease on the population. Moreover, higher γ values lead to a quicker attainment of a steady-state, where the number of new infections is roughly equal to the number of recoveries. This could potentially shorten the duration of outbreaks, allowing healthcare systems to recover more quickly and reduce long-term societal and economic impacts.

Investigate how the spread of the disease changes as I_0 varies from 1 to 100

For this investigation, the initial number of infected individuals I_0 was varied from 1 to 100. The values used were $I_0 = 1, 10, 25, 50, 100$.

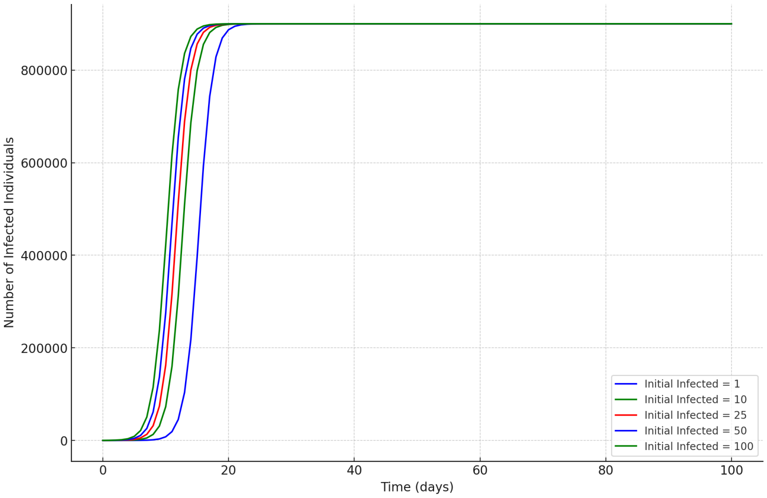

Figure 3: Effect of varying I_0 on the SIS model

The initial number of infected individuals (I_0) has a more pronounced impact on short-term dynamics. Higher I_0 values lead to a faster initial spread due to a larger pool of infected individuals. As the disease progresses and a significant portion of the population becomes infected, the rate of transmission declines. This results in the infection reaching similar prevalence levels over time, irrespective of the initial I_0 value. Ultimately, the long-term dynamics are influenced more by transmission and recovery rates than the initial I_0. This observation emphasizes the need for effective strategies to control transmission rates and optimize recovery efforts, as these factors play a pivotal role in managing the outbreak's impact on the population.

Equilibrium Points and Their Stability

Equilibrium points occur where the system remains stable, i.e., where $\frac{dS}{dt} = 0$ and $\frac{dI}{dt} = 0$.

From the first equation, setting $\frac{dS}{dt} = 0$:

$$-\beta SI + \gamma I = 0$$

$$\Rightarrow \beta SI = \gamma I$$

$$\Rightarrow \beta S = \gamma \quad \text{or} \quad I = 0.$$

From the second equation, setting $\frac{dI}{dt} = 0$:

$$\beta SI - \gamma I = 0$$

$$\Rightarrow \beta S = \gamma \quad \text{or} \quad I = 0.$$

Combining these, two equilibrium points emerge:

1. $(S, I) = (1, 0)$ — Disease-free equilibrium.
2. $(S, I) = \left(\frac{\gamma}{\beta}, 1 - \frac{\gamma}{\beta}\right)$ where $I \neq 0$ — Disease-Persistent equilibrium.

Stability Analysis

To assess the stability of these points, a linear stability analysis is performed using the Jacobian matrix J of the system.

$$J = \begin{pmatrix} -\beta I & \gamma - \beta S \\ \beta I & \beta S - \gamma \end{pmatrix}$$

Stability at Disease-free Equilibrium $(S, I) = (1, 0)$

For this equilibrium, the Jacobian matrix simplifies to:

$$J = \begin{pmatrix} 0 & \gamma \\ 0 & -\gamma \end{pmatrix}$$

The eigenvalues are $\lambda_1 = \gamma$ and $\lambda_2 = -\gamma$.

- $\lambda_1 = \gamma$ (usually $\gamma > 0$) — One eigenvalue is positive, making the disease-free equilibrium unstable.

Stability at Disease-Persistent Equilibrium $(S, I) = \left(\frac{\gamma}{\beta}, 1 - \frac{\gamma}{\beta}\right)$

For this equilibrium, the Jacobian matrix becomes:

$$J = \begin{pmatrix} -\gamma + \beta & 0 \\ \gamma - \beta & 0 \end{pmatrix}$$

The eigenvalues are $\lambda_1 = 0$ and $\lambda_2 = -\beta + \gamma$.

- $\lambda_1 = 0$ — The system is neutrally stable along this direction.
- $\lambda_2 = -\beta + \gamma$ — This eigenvalue will be negative if $\beta > \gamma$, indicating that the equilibrium is stable in this direction.

Conditions for an Epidemic

1. **Epidemic occurs**: An epidemic will occur if the disease-free equilibrium is unstable, which is the case here as one of its eigenvalues is positive ($\gamma > 0$).
2. **Disease does not reach epidemic proportions**: The disease-persistent equilibrium is neutrally stable, meaning that small perturbations will not cause it to move away from this point. Thus, in this particular SIS model, if the system reaches the disease-persistent equilibrium, it will stay there, and the disease will not reach epidemic proportions.

Cyclic Behavior Possibility

In the standard Susceptible-Infected-Susceptible (SIS) model, cyclic behavior involving alternating periods of high and low disease prevalence is not inherently present. The model is governed by first-order ordinary differential equations (ODEs) that are linear in each compartment. Systems described by such linear equations typically do not manifest complex dynamical behaviors like oscillations or cycles unless influenced by external factors or non-linearities.

The stability analysis of the SIS model identifies two primary equilibrium states: a Disease-Free Equilibrium at $(S, I) = (1, 0)$, which is found to be unstable, and a Disease-Persistent Equilibrium at $(S, I) = \left(\frac{\gamma}{\beta}, 1 - \frac{\gamma}{\beta}\right)$ where $I \neq 0$, which is neutrally stable. Notably, the eigenvalues associated with these equilibrium points are real numbers, not complex, further corroborating that the system lacks intrinsic oscillatory behavior.

Moreover, the basic SIS model does not account for complexities such as time delays, seasonal variations in transmission rates, or behavioral changes in the population—factors that could potentially introduce cyclical fluctuations in disease prevalence. Therefore, within the framework of the standard SIS model, the occurrence of cyclic behavior in disease prevalence is not anticipated.

Conclusion

The transition from the Susceptible-Infected-Recovered (SIR) model to the Susceptible-Infected-Susceptible (SIS) model was necessitated by new clinical evidence that challenged the long-standing assumption of immunity post-recovery. Through rigorous simulation, the study illuminated how varying key parameters like the infectious contact rate (β), the recovery rate (γ), and the initial number of infected individuals (I_0) could impact the spread of the disease. Notably, higher β values resulted in accelerated disease transmission, emphasizing the need for strategies to control this rate. Conversely, an increase in γ led to quicker recoveries and fewer peak infections, highlighting its role in effective public health strategies. The initial number of infected individuals (I_0) influenced short-term dynamics but had less impact on long-term outcomes.

Stability analysis further revealed that the disease-free equilibrium is inherently unstable, making an epidemic likely under the current conditions. On the other hand, the disease-persistent equilibrium was found to be neutrally stable, suggesting that the disease, once established, is likely to persist within the population. These findings have significant implications for public health policy, emphasizing the importance of reducing infectious contact rates and optimizing recovery efforts. The study serves as a cornerstone for more complex models like SIRS or SEIR, which could capture cyclical patterns and additional complexities not accounted for in the basic SIS model. In summary, the adapted SIS model provides a robust framework for understanding the disease's potential evolution, thereby offering valuable guidance for devising effective containment and treatment strategies.

Simulation Code

Application of Runge-Kutta Equation

```
# Import required libraries
import numpy as np
import matplotlib.pyplot as plt

# Define the SIS model using Runge-Kutta 4th Order Method
def sis_model(Y, t, beta, gamma):
    S, I = Y
    dS = -beta * S * I + gamma * I
    dI = beta * S * I - gamma * I
    return [dS, dI]

# Implementing 4th Order Runge-Kutta Method
def runge_kutta_4th(func, Y0, t, args=()):
    h = t[1] - t[0]
    N = len(t)
    Y = np.empty((N, len(Y0)))
    Y[0] = Y0
    for i in range(0, N - 1):
        k1 = np.multiply(func(Y[i], t[i], *args), h)
        k2 = np.multiply(func(Y[i] + 0.5 * k1, t[i] + 0.5 * h, *args), h)
        k3 = np.multiply(func(Y[i] + 0.5 * k2, t[i] + 0.5 * h, *args), h)
        k4 = np.multiply(func(Y[i] + k3, t[i] + h, *args), h)
        Y[i + 1] = Y[i] + (k1 + 2 * k2 + 2 * k3 + k4) / 6
    return Y

# Initial parameters
N = 1_000_000  # Total population
I0 = 1 / N  # Initial proportion of infected individuals
S0 = 1 - I0  # Initial proportion of susceptible individuals
gamma = 1 / 100  # Recovery rate (duration an individual remains infected is 100 days)
Y0 = [S0, I0]  # Initial conditions vector
t = np.linspace(0, 100, 100)  # Time grid (100 days)
colors = [’b’, ’g’, ’r’, ’c’, ’m’]  # Colors for the plots

# Different beta values for investigation
beta_values_to_investigate = [0.1, 0.3, 0.5, 0.7, 1]

# Create a figure for LaTeX without a title
plt.figure(figsize=(12, 8))

# Loop through different beta values
for idx, beta in enumerate(beta_values_to_investigate):
    solution = runge_kutta_4th(sis_model, Y0, t, args=(beta, gamma))
    S, I = solution.T
    I_abs = I * 1_000_000  # Convert to absolute numbers
    plt.plot(t, I_abs, label=f’ = {beta}’, color=colors[idx % len(colors)])

# Add labels
plt.xlabel(’Time (days)’)
plt.ylabel(’Number of Infected Individuals’)
plt.legend()
plt.grid(True)
```

plot_filepath

Gamma Simulation for Investigation

```
# Different gamma values for investigation
gamma_values_to_investigate = [0.01, 0.03, 0.05, 0.07, 0.1]

# Create a figure for LaTeX without a title
plt.figure(figsize=(12, 8))

# Loop through different gamma values
for idx, gamma in enumerate(gamma_values_to_investigate):
    solution = runge_kutta_4th(sis_model, Y0, t, args=(beta, gamma))
    S, I = solution.T
    I_abs = I * 1_000_000  # Convert to absolute numbers
    plt.plot(t, I_abs, label=f’ = {gamma}’, color=colors[idx % len(colors)])

# Add labels
plt.xlabel(’Time (days)’)
plt.ylabel(’Number of Infected Individuals’)
plt.legend()
plt.grid(True)
```

gamma_plot_filepath

Initial Infected Individuals Simulation for Investigation

```
    # Varying initial infected individuals from 1 to 100
initial_infected_values_abs = [1, 10, 25, 50, 100]

# Create a figure
plt.figure(figsize=(12, 8))
```

```
# Loop through different initial infected individuals
for idx, I0_abs in enumerate(initial_infected_values_abs):
    # Normalizing to proportions
    I0 = I0_abs / 1_000_000
    S0 = 1 - I0  # Since S + I = 1
    Y0 = [S0, I0]

    solution = runge_kutta_4th(sis_model, Y0, t, args=(beta, gamma))
    S, I = solution.T
    I_abs = I * 1_000_000  # Convert to absolute numbers
    plt.plot(t, I_abs, label=f'Initial Infected = {I0_abs}', color=colors[idx % len(colors)]

# Add titles and labels
plt.title('Effect of Initial Infected Individuals on SIS Model (Runge-Kutta 4th Order Metho
plt.xlabel('Time (days)')
plt.ylabel('Number of Infected Individuals')
plt.legend()
plt.grid(True)

# Show the plot
plt.show()
```

The outputs are the placed graphs