Uncertainty Quantification (ACM41000)
Mini Project 1

Ian Towey

04128591

June 19, 2018

Abstract

Analytic and numerical analysis of a basic SIR epidemic model

Contents

0 Introduction

This report presents analysis of a SIR epidemic model on a fixed population of size N.

The population is split into 3 groups susceptible, infected and recovered, $x(t)$, $y(t)$ and $z(t)$ respectively.

$$x(t) + y(t) + z(t) = N \tag{1}$$

The set of equations to model the different population subgroups are:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = -kxy \tag{2a}$$

$$\frac{\mathrm{d}y}{\mathrm{d}t} = kxy - ly \tag{2b}$$

$$\frac{\mathrm{d}z}{\mathrm{d}t} = ly \tag{2c}$$

1 Analytic analysis x- and y- equations alone

From Equation (2a)

$$y = \frac{1}{l}\frac{\mathrm{d}z}{\mathrm{d}t} \tag{3}$$

Substituting Equation (3) into Equation (2a) and solving for $x(t)$:

$$\begin{aligned}
\frac{\mathrm{d}x}{\mathrm{d}t} &= -\frac{kx}{l}\frac{\mathrm{d}z}{\mathrm{d}t} \\
\frac{\mathrm{d}x}{x} &= -\frac{k\mathrm{d}z}{l} \\
\int \frac{\mathrm{d}x}{x} &= -\frac{k}{l}\int \mathrm{d}z \\
\ln(x) &= -\frac{kz}{l} + C \\
x(t) &= e^{-\frac{kz}{l}}e^{C}
\end{aligned}$$

Letting $e^C = x_0$ gives

$$x(t) = x_0 e^{-\frac{kz}{l}} \tag{4}$$

2 Solve fot y(t)

$$\begin{aligned}
\frac{\mathsf{d}y}{\mathsf{d}t} &= kxy - ly \\
\frac{\mathsf{d}y}{\mathsf{d}t} &= -\left(\frac{\mathsf{d}x}{\mathsf{d}t} + \frac{\mathsf{d}z}{\mathsf{d}t}\right) \\
\mathsf{d}y &= -\left(\mathsf{d}x + \mathsf{d}z\right) \\
\int \mathsf{d}y &= -\left(\int \mathsf{d}x + \int \mathsf{d}z\right) \\
y(t) &= -x(t) - z(t) + C
\end{aligned}$$

From Equation (1), we can see that $C = N$, giving:

$$y(t) = -x(t) - z(t) + N$$

Substituting RHS of Equation (6) for $x(t)$ yields:

$$y(t) = N - x_0 e^{-\frac{kz}{l}} - z(t) \tag{5}$$

3 Quick view of $\frac{\mathsf{d}z}{\mathsf{d}t}$

$$\frac{\mathsf{d}z}{\mathsf{d}t} = ly$$

Substituting RHS of Equation (7) in for $y(t)$, yields:

$$\frac{\mathsf{d}z}{\mathsf{d}t} = l[N - x_0 e^{-\frac{kz}{l}} - z(t)] \tag{6}$$

Equation (6) is an ODE with a single dependent variable z and 4 independent constants , l, k, N, x_0. While this could probably be analyzed using qualitative analysis such as finding fixed points and testing the stability of these fixed points, doing so would not be easy due to the number of independent constants and their impact on this analysis.

4 Rescaling to reduce the number of constants

Using variable transforms

(i) $u = \frac{kz}{l}$; (ii) $a = \frac{N}{x_0}$; (iii) $b = \frac{l}{kx_0}$; (iv) $\tau = kx_0 t$

$$\begin{aligned}
\frac{\mathsf{d}z}{\mathsf{d}t} &= l[N - x_0 e^{-\frac{kz}{l}} - z(t)] \\
\frac{1}{x_0}\frac{\mathsf{d}z}{\mathsf{d}t} &= l[\frac{N}{x_0} - e^{-\frac{kz}{l}} - \frac{z}{x_0}] \\
\frac{1}{x_0}\frac{\mathsf{d}z}{\mathsf{d}t} &= l[a - e^{-\frac{kz}{l}} - \frac{z}{x_0}]
\end{aligned}$$

From (i) above, $z = \frac{ul}{k}$:

$$\frac{1}{x_0}\frac{\mathsf{d}z}{\mathsf{d}t} = l[a - e^{-\frac{\cancel{k}u}{\cancel{k}}} - \frac{ul}{kx_0}] \tag{7}$$

From (iii) $b = \frac{l}{kx_0}$:

$$\frac{1}{x_0}\frac{\mathsf{d}z}{\mathsf{d}t} = l[a - e^{-u} - bu] \tag{8}$$

$$u = \frac{kz}{l} \Longrightarrow \mathsf{d}u = \frac{k\mathsf{d}z}{l} \Longrightarrow \mathsf{d}z = \frac{l\mathsf{d}u}{k}$$

$$\tau = kx_0t \Longrightarrow \mathsf{d}\tau = kx_0\mathsf{d}t \Longrightarrow \mathsf{d}t = \frac{\mathsf{d}\tau}{kx_0}$$

Using the above definitions for $\mathsf{d}z$ and $\mathsf{d}t$ to find ratio of $\frac{\mathsf{d}z}{\mathsf{d}t}$

$$\frac{\mathsf{d}z}{\mathsf{d}t} = \frac{l\mathsf{d}u}{k}\frac{1}{\frac{\mathsf{d}\tau}{kx_0}} = \frac{l\mathsf{d}u}{\cancel{k}}\frac{\cancel{k}x_0}{\mathsf{d}\tau} = lx_0\frac{\mathsf{d}u}{\mathsf{d}\tau}$$

Substituting for $\frac{\mathsf{d}z}{\mathsf{d}t}$ into Equation (8):

$$\frac{1}{\cancel{x_0}}\cancel{lx_0}\frac{\mathsf{d}u}{\mathsf{d}\tau} = \cancel{l}[a - e^{-u} - bu]$$

Yields the desired equation

$$\frac{\mathsf{d}u}{\mathsf{d}\tau} = a - e^{-u} - bu \tag{9}$$

5 Fixed points / stability analysis of $\frac{\mathsf{d}u}{\mathsf{d}\tau}$

$$\frac{\mathsf{d}u}{\mathsf{d}\tau} = a - e^{-u} - bu$$

$$f(u) = a - e^{-u} - bu$$

Analyzing $f(u)$ at zero, $f(0) = a - e^0 - b0 = a - 1 \geq 0$,

- if $a = 1$, then $f(0) = 0$, this is the trivial fixed point which represented when there are no infected people in the system initially, as $a = N/x_0 = (x_0 + y_0)/x_0 = 1$, therefore N = x_0
- if $a > 0$, then $f(0) > 0$, therefore $a = N/x_0 = (x_0+y_0)/x_0 > 1$ and $y_0 > 0$ indicates there are some infected people in the system at t_0.

Analyzing $f(u)$ as $u \to \infty$

$$\lim_{u\to\infty} f(u) = \lim_{u\to\infty}[a - e^{-u} - bu] = -bu \tag{10}$$

- As u grows large the linear terms $-bu$ dominates and the equation looks like a stright line as show in Figure (2).

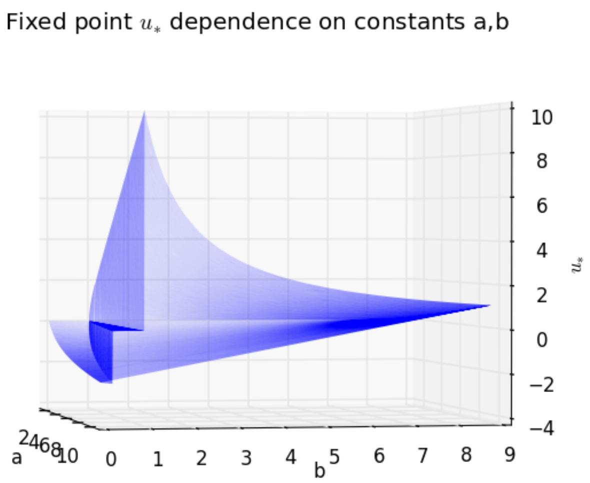

Figure 1: u_* vs a vs b

Figure (1) shows a 3D plot of u_* for a range of values for a and b,

6 Analysis of parameter b

From Equation (10), we can see the value of parameter b dominates for large values of u. To find the single maximum analytically we solve $f'(u_*) = 0$

$$f'(u_*) = e^{-u_*} - b = 0$$

$$u_* = -ln(b)$$

Checking the sign of the second derivative $f''(u) = -e^{-u} < 0$ for all u , so $u_* = -ln(b)$ is a maximum. $f(u)$ has one fixed point u_*, and from Figure (2) and (3) this is a stable fixed point.

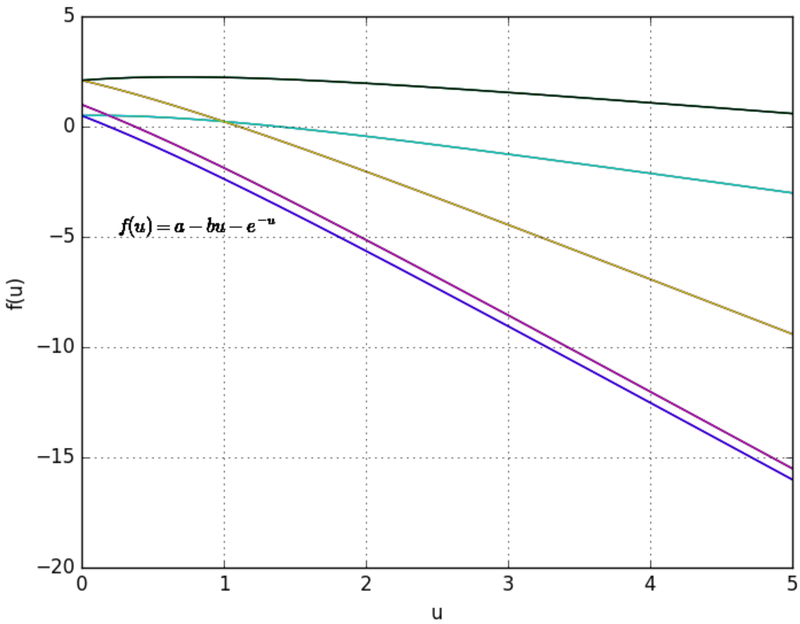

Figure 2: Fixed point U_*, for different values of a and b.

Figure (3) shows a screen shot of the interactive *maxima* plot with sample trajectory paths (these can be generated by just tapping on the plot with cursor and using the sliders to chnage the values of a and b)

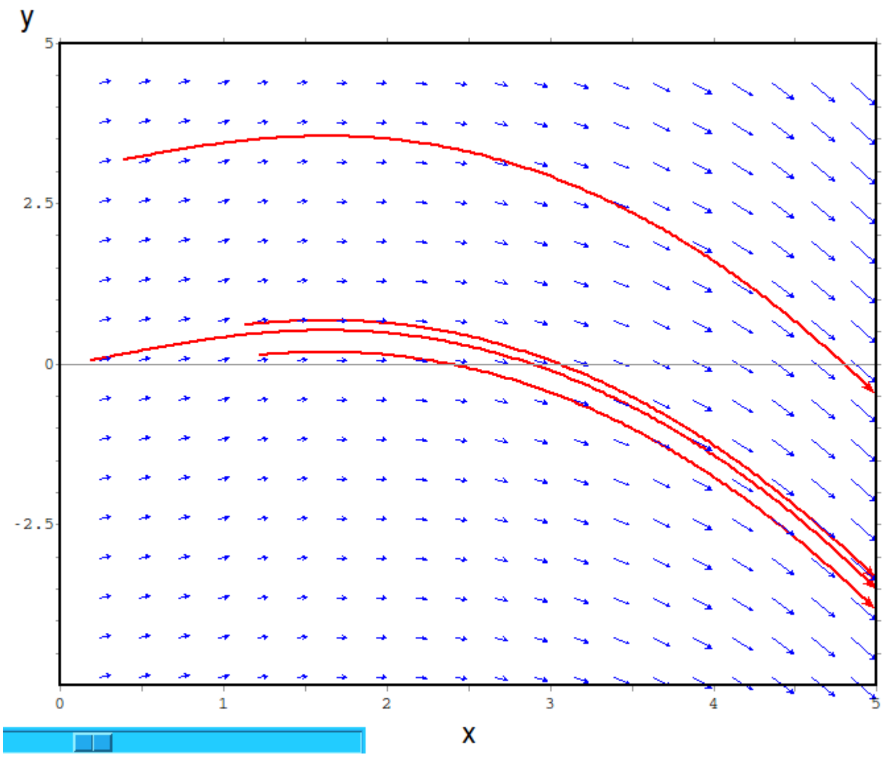

Figure 3: direction field of $f(u)$

7 Numerically solutions

The Python package *scipy* and specifically the *integrate* function were used to solve the Equation (11) numerically for various values of a and b. Figures 3 - 8 graphically show the evolution of the infections for a few representative values of a and b.

Population N = 1000				
	a	b	x_0	y_0
Figure (3)	1.5	0.9	6666.66	3333.33
Figure (4)	2	3.5	5000	5000
Figure (5)	3.1	2.5	3225.8	6774.19
Figure (6)	3.1	0.5	3225.8	6774.19
Figure (7)	1.5	3.5	6666.66	3333.33
Figure (8)	1.1	0.01	9090.9	909.09

From the plots the significance of the parameter b is evident. For values of $b < 1$, the number of infected increases small values of τ before decaying to 0, with the decay rate increasing as b gets larger in size. When $b > 1$, the number of infected decreases forall τ.

The smaller the value of b the faster the rate if infection is and the slower the rate of recovery is as can be seen in Figure 8.

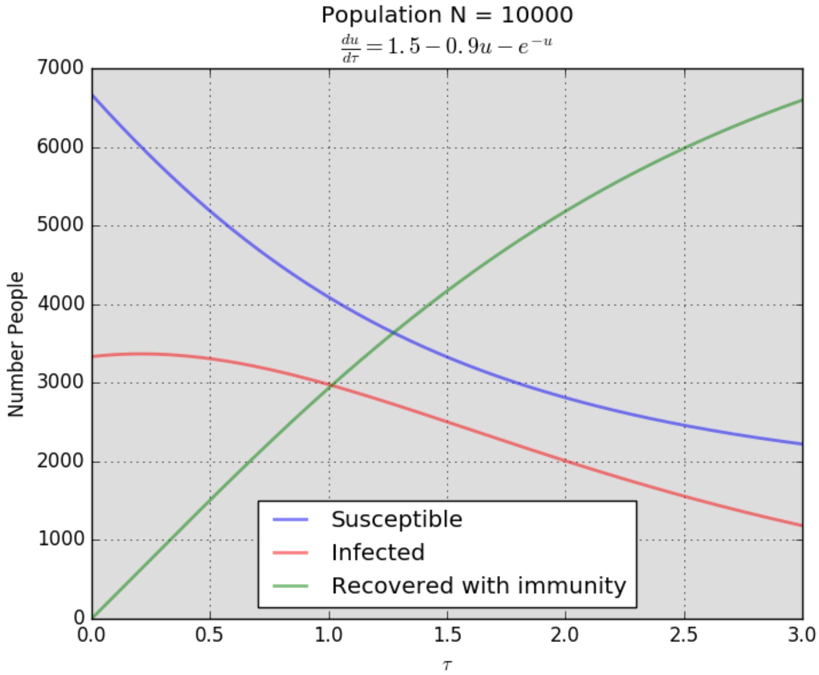

Figure 4: Todo

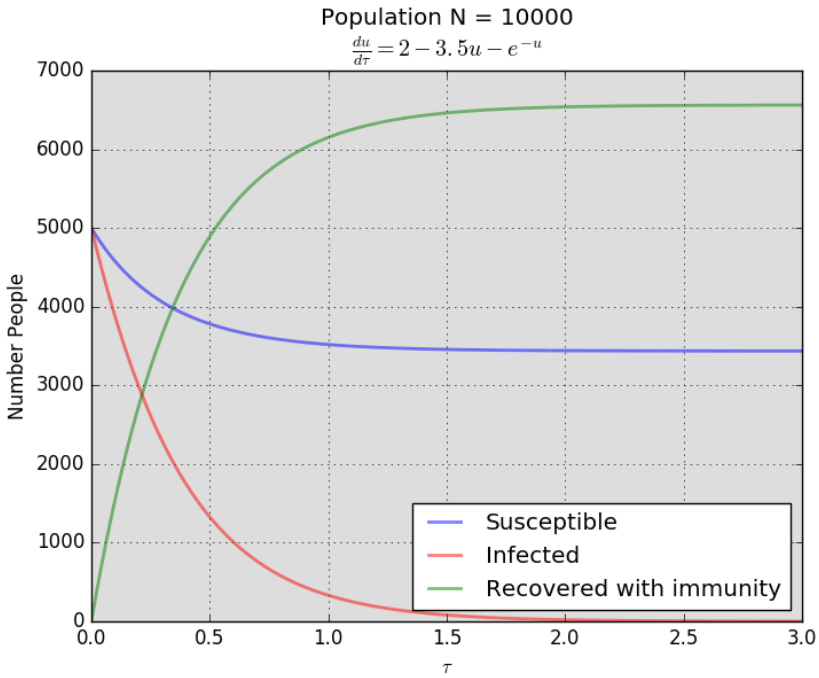

Figure 5: Todo

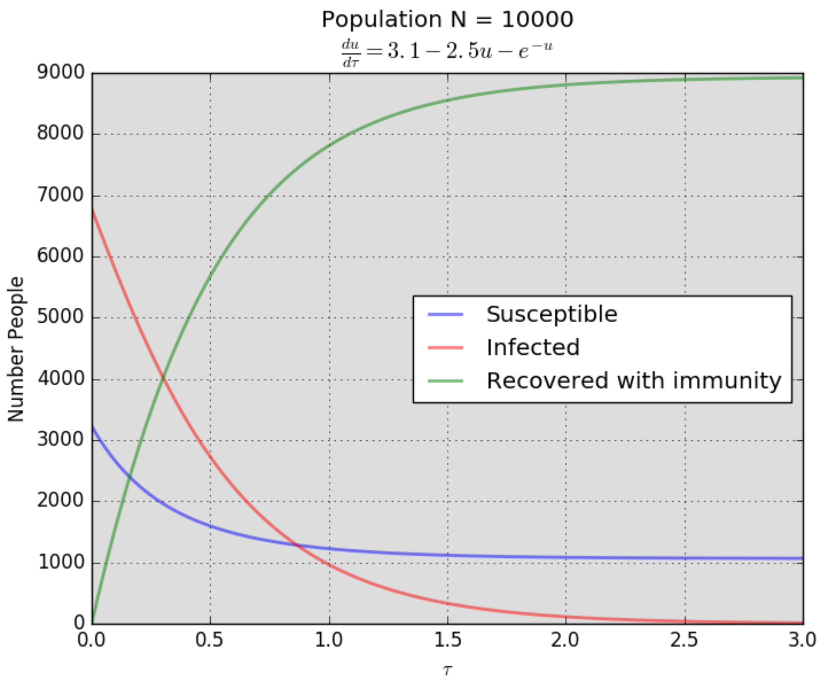

Figure 6: Todo

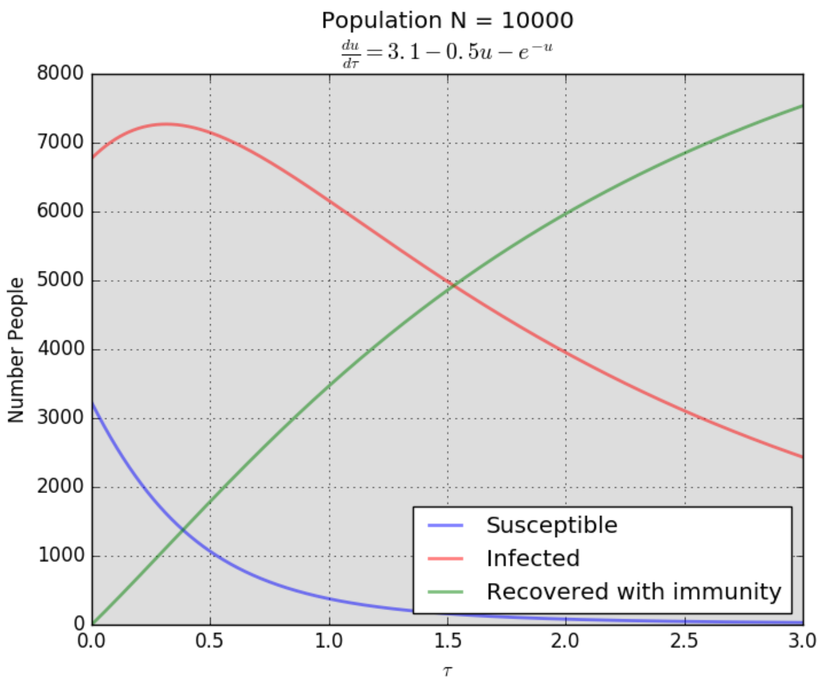

Figure 7: Todo

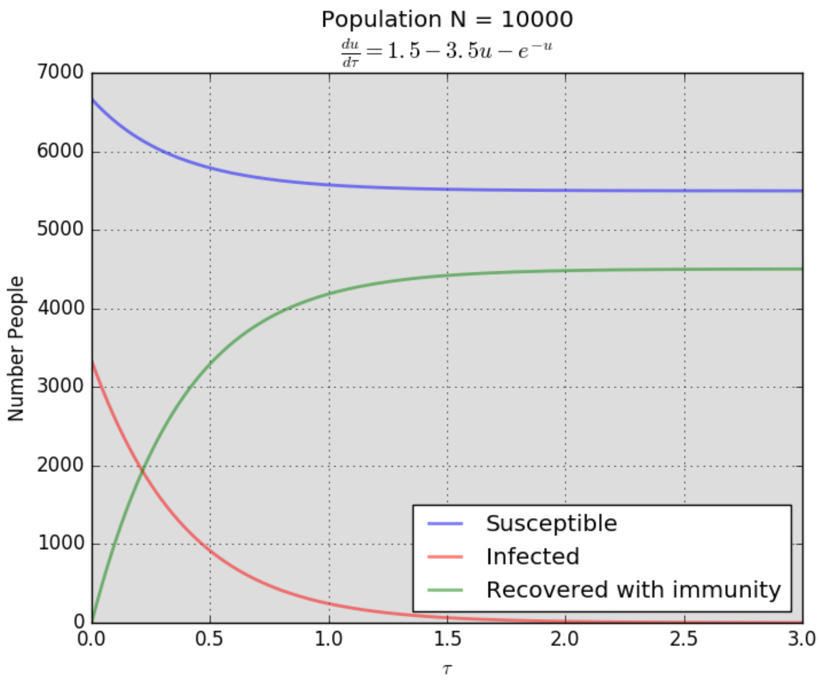

Figure 8: Todo

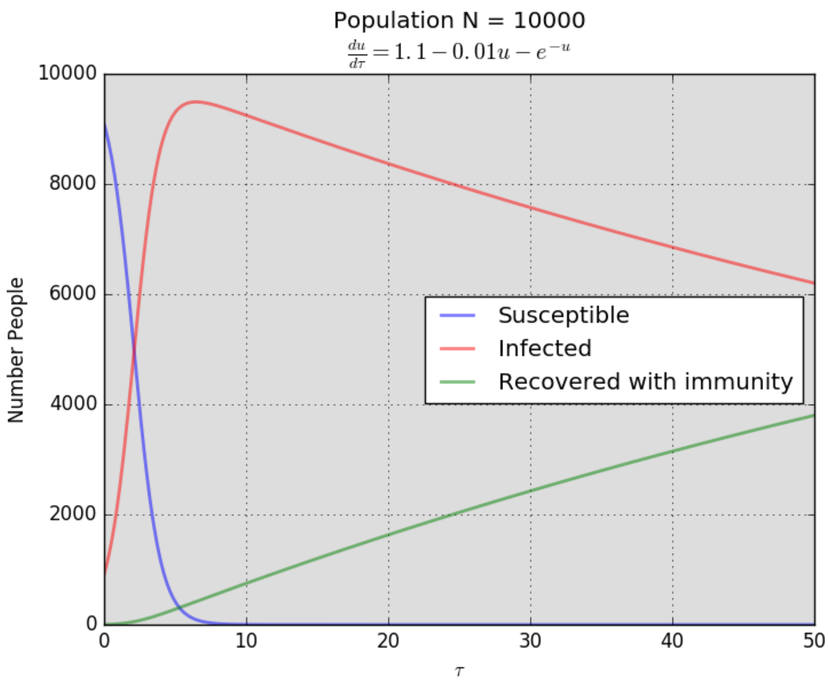

Figure 9: Todo

8 Full SIR model

$$
\begin{aligned}
F(x) &= -kxy + mz \\
G(y) &= kxy + ly \\
H(z) &= ly - mz
\end{aligned}
\tag{11}
$$

8.1 Show $N = x + y + z$ is a constant

$$
\begin{aligned}
\frac{\mathsf{d}N}{\mathsf{d}t} &= \frac{\mathsf{d}x}{\mathsf{d}t} + \frac{\mathsf{d}y}{\mathsf{d}t} + \frac{\mathsf{d}z}{\mathsf{d}t} \\
0 &= -kxy + mz + kxy + ly + ly - mz \\
0 &= 0
\end{aligned}
$$

8.2 Show $x = N, y = z = 0$ is fixed point

Substituting $x = N, y = z = 0$ in the Equations (2,3,4), yields

$$
\frac{\mathsf{d}x}{\mathsf{d}t} = \frac{\mathsf{d}y}{\mathsf{d}t} = \frac{\mathsf{d}z}{\mathsf{d}t} = 0 \tag{12}
$$

8.3 Compute the Jacobian with the fixed point, Compute the eigenvalues of the Jacobian

The jacobian associated with this system is defined as

$$
J = \begin{bmatrix} \frac{\partial F_x}{\partial t} & \frac{\partial F_y}{\partial t} & \frac{\partial F_z}{\partial t} \\ \frac{\partial G_x}{\partial t} & \frac{\partial G_y}{\partial t} & \frac{\partial G_z}{\partial t} \\ \frac{\partial H_x}{\partial t} & \frac{\partial H_y}{\partial t} & \frac{\partial H_z}{\partial t} \end{bmatrix} = \begin{bmatrix} -ky & -kx & m \\ ky & kx - l & m \\ 0 & l & -m \end{bmatrix}
$$

Evaluating the jacobian at the fixed point $x = N, y = z = 0$

$$
J_* = \begin{bmatrix} 0 & -kN & m \\ 0 & kN - l & m \\ 0 & l & -m \end{bmatrix}
$$

The eigenvalues in this case are the diagonal of J_*

$$
\begin{aligned}
\lambda_1 &= 0 \\
\lambda_2 &= kN - l \\
\lambda_3 &= -m
\end{aligned}
$$

8.4 Fixed point stability

- $\lambda_1 = 0$ has no impact on the stability since it is neutral
- $\lambda_3 = -m$ is stable as $m > 0$
- $\lambda_2 = kN - l$ is stable if $kN < l$ and unstable if $kN > l$

For the fixed point to be stable, the infection rate k would need to be very small relative to the recovery rate l

Appendices

A Python Code

Python used to generate Figure (1)

```
path = '/home/ian/Desktop/ACM41000-Uncertainty-Quantification/
    assignment1/'

from scipy import optimize
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

def f(u, a, b):
    return a - b*u[0] - exp(-u[0])

step = 0.01
a_p = np.arange(1 + step, 10 + step, step)
b_p = np.arange(0 + step, 9 + step, step)
a_ij = []
b_ij = []
u_ij = []

for i in range(0,len(a_p)):
    for j in range(0,len(b_p)):
        sol = optimize.root(fun = f,x0 = [0], args = (a_p[i],b_p
            [j]), method = 'hybr')
        a_ij.append(a_p[i])
        b_ij.append(b_p[j])
        u_ij.append(sol.x[0])

fig = plt.figure()
ax = fig.gca(projection='3d')
ax.plot(a_ij, b_ij, u_ij, lw=step)
ax.set_xlabel("a")
ax.set_ylabel("b")
ax.set_zlabel(r'$u_{*}$')
ax.set_title("Fixed point " + r'$u_{*}$' + " dependence on
    constants a,b")
plt.show()
plt.savefig(path + '3D_plot.png')
```

Python used to generate Figure (2)

```
path = '/home/ian/Desktop/ACM41000-Uncertainty-Quantification/
    assignment1/'

from scipy import optimize
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

def f(u, a, b):
    return a - b*u[0] - exp(-u[0])

U = np.linspace(start = 0, stop = 5, num = 10000, endpoint=True)
for param in [((1.5,0.9),'1', 10),((2,3.5),'2',10),((3.1,2.5),'3
    ',10),((3.1,0.5),'4',10),((1.5,3.5),'5',10),]:
    a,b = param[0]
    file_idx = param[1]
    u_max = param[2]
```

```
    F_U = [f([u],a,b) for u in U]
    plt.plot(U, F_U)
    plt.grid(True)
    plt.text(0.25, -4.8, r'$f(u) = a - bu - e^{-u}$')
    plt.xlabel('u')
    plt.ylabel('f(u)')
    plt.show()
plt.savefig(path + 'fp_diff_ab.png')
```

Python used to numerically solve the system of equations and generate Figures (4-9)

```
import matplotlib.pyplot as plt
from scipy import integrate

#function def
def du_dt(u,t,a,b):
    return a - b*u - exp(-u)

path = '/home/ian/Desktop/ACM41000-Uncertainty-Quantification/
    assignment1'
#population size
N = 10000
for param in [((1.5,0.9),'1', 3),((2,3.5),'2',3),((3.1,2.5),'3'
    ,3),((3.1,0.5),'4',3),((1.5,3.5),'5',3),((1.1,0.01),'6',50)
    ]:
    a, b = param[0]
    file_idx = param[1]
    tau_max = param[2]
    u_step = linspace(0, tau_max,  10001)
    #solve ode
    U, infodict = integrate.odeint(du_dt, y0 = 0, t = u_step,
        args = (a,b), full_output=True)
    if infodict['message'] != 'Integration successful':
        #rescale x,y,z reletive to the population size
        x_x0 = exp(-U.T[0]) * N / a
        y_y0 = (a - b*U.T[0] - exp(-U.T[0])) * N / a
        z_z0 = (b*U.T[0]) * N / a
        #plot x_x0, y_y0, z_z0
        fig = plt.figure(facecolor='w')
        ax = fig.add_subplot(111, axis_bgcolor='#dddddd',
            axisbelow=True)
        ax.plot(u_step, x_x0, 'b', alpha=0.5, lw=2, label='
            Susceptible')
        ax.plot(u_step, y_y0, 'r', alpha=0.5, lw=2, label='
            Infected')
        ax.plot(u_step, z_z0, 'g', alpha=0.5, lw=2, label='
            Recovered with immunity')
        ax.set_xlabel(r'$\tau$')
        ax.set_ylabel('Number People')
        ax.legend(loc='best')
        plt.title( 'Population N = 10000\n'+ r'$\frac{du}{d\tau}
             = ' + str(a) + ' - ' + str(b) + 'u -e^{-u}$')
        plt.legend(loc='best')
        plt.grid()
        plt.show()
        plt.savefig(path + '/q7_' + file_idx+ '.png')
    else:
        print 'error (a = {}, b = {})'.format(a, b)
```

B Maxima Code

Maxima command used to generate Figure (3)

```
plotdf(
        [a - b*x - exp(-x)],
        [parameters,"a=1.2,b=0.5"],
        [sliders,"a=1:3:,b=0:5"],
        [tstep,0.1],
        [x,0,5],
        [y,-5,5],
        [direction,forward]
);
```