



This demonstrates a typical Epidemic curve. To flatten the curve the following measures should be taken:

1. Social distancing
2. Wearing Mask
3. Wash hands frequently
4. Avoid touching the face
5. Cover cough or sneezes with a tissue
6. Identify and isolate individuals who have come into contact with an infected person
7. Improve testing capacity to identify infected individuals early
8. Implement restrictions on non-essential social activities
9. Avoid large gatherings

**Matlab code:**

function dydt = diff\_eq(~, y, alpha, beta, gamma, sigma)

S = y(1); E = y(2); I = y(3); R = y(4);

dSdt = -beta\*S\*I + gamma\*R;

dEdt = beta\*S\*I - sigma\*E;

dIdt = sigma\*E - alpha\*I;

dRdt = alpha\*I - gamma\*R;

dydt = [dSdt; dEdt; dIdt; dRdt];

end

clc;

close all;

Clear;

beta = 0.32;

sigma = 0.87;

alpha = 0.15;

gamma = 0.69;

S0 = 9785;

E0 = 115;

I0 = 100;

R0 = 0;

tspan = 0:0.01:100;

[t, y] = ode45(@(t,y)diff\_eq(t, y, alpha, beta, gamma, sigma), tspan, [S0; E0; I0; R0]);

S = y(:, 1);

E = y(:, 2);

I = y(:, 3);

R = y(:, 4);

plot(t, S, 'b-', t, E, 'r-', t, I, 'g-', t, R, 'm-');

xlabel('Time');

ylabel('Population');

legend('Susceptible', 'Exposed', 'Infectious', 'Recovered');

**Python code:**

import numpy as np

from scipy.integrate import odeint

import matplotlib.pyplot as plt

def seirs\_model(y, t, alpha, beta, gamma, sigma):

    S, E, I, R = y

    dSdt = -beta\*S\*I + gamma\*R

    dEdt = beta\*S\*I - sigma\*E

    dIdt = sigma\*E - alpha\*I

    dRdt = alpha\*I - gamma\*R

    return [dSdt, dEdt, dIdt, dRdt]

*# Define parameters*

beta = 0.32

sigma = 0.87

alpha = 0.15

gamma = 0.69

*# Set initial conditions*

S0 = 9785

E0 = 115

I0 = 100

R0 = 0

*# Set time span*

tspan = np.arange(0, 100, 0.01)

*# Solve the differential equations*

y0 = [S0, E0, I0, R0]

sol = odeint(seirs\_model, y0, tspan, args=(alpha, beta, gamma, sigma))

*# Extract solutions*

S, E, I, R = sol.T

*# Plot the results*

plt.plot(tspan, S, label='Susceptible')

plt.plot(tspan, E, label='Exposed')

plt.plot(tspan, I, label='Infectious')

plt.plot(tspan, R, label='Recovered')

plt.xlabel('Time')

plt.ylabel('Population')

plt.legend()

plt.show()