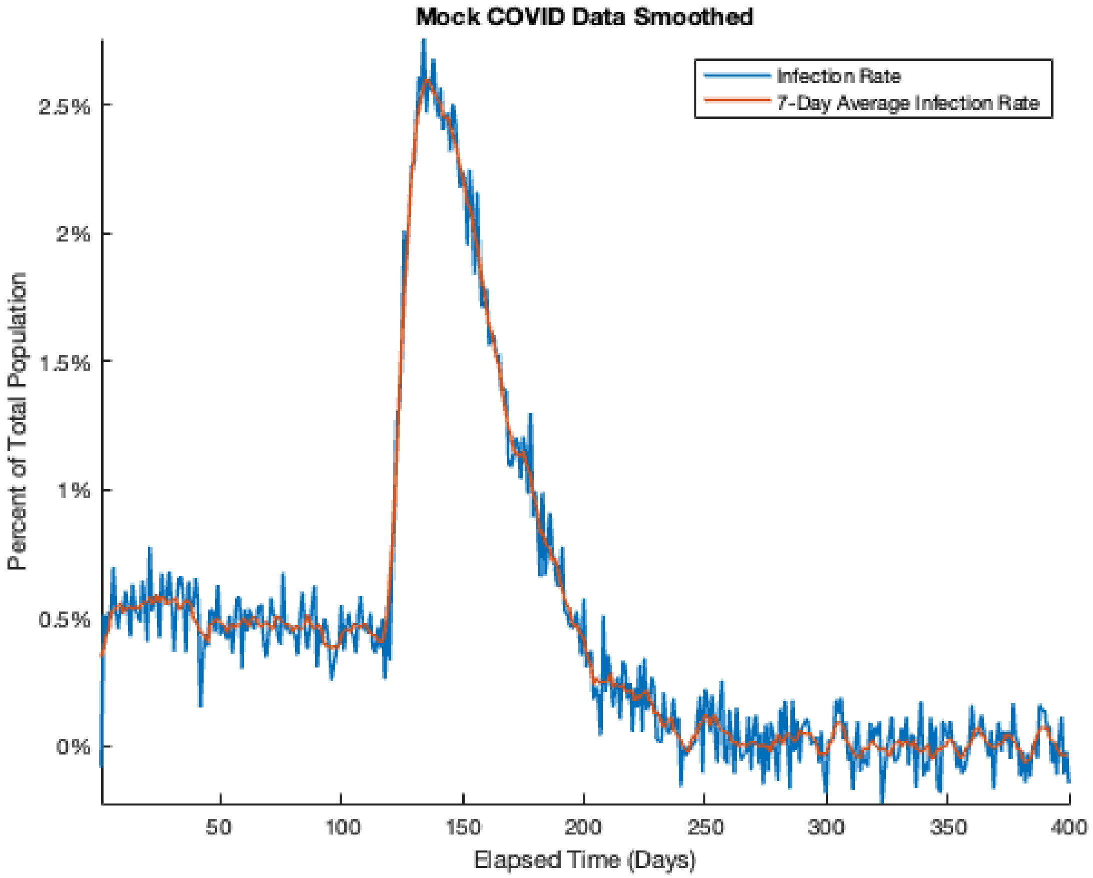
```
% Part IV: Modeling Vaccination & Competition Challenge

load('mockdata2023.mat');

newDeaths = cumulativeDeaths(1);
for j = 2:length(cumulativeDeaths)
    newDeaths = [newDeaths, cumulativeDeaths(j) - cumulativeDeaths(j - 1)];
end

% smooth out the curve (7-day average)
smoothNewInfections = [mean(newInfections(1:4)), ...
    mean(newInfections(1:5)), mean(newInfections(1:6))];
for j = 4:length(newInfections) - 4
    smoothNewInfections = [smoothNewInfections, ...
        mean(newInfections(j - 3:j + 3))];
end
smoothNewInfections = [smoothNewInfections, ...
    mean(newInfections(length(newInfections) ...
    - 5:length(newInfections))), ...
    mean(newInfections(length(newInfections) ...
    - 4:length(newInfections))), ...
    mean(newInfections(length(newInfections) - 3:length(newInfections)))];

hold on;
plot(newInfections * 100, 'LineWidth', 1.5);
plot(smoothNewInfections * 100, 'LineWidth', 1.5);
hold off;
axis tight;
title('Mock COVID Data Smoothed');
legend('Infection Rate', '7-Day Average Infection Rate');
xlabel('Elapsed Time (Days)');
ylabel('Percent of Total Population');
ytickformat('percentage');
exportgraphics(gca, 'mock_data.png');

% check for sharp decline in infection rate to find the beginning of
% vaccinations
for j = 1:length(smoothNewInfections) - 30
    if issorted(smoothNewInfections(j:j + 30), 'descend')
        vaxDate = j;
        break
    end
end
% check for convergence in infection rate to find the beginning of
% breakthrough infections
for j = vaxDate:length(smoothNewInfections) - 30
    if mean(smoothNewInfections(j:j + 30) <= 0.001)
        breakthroughDate = j;
        break
    end
end
vaxRate = (smoothNewInfections(vaxDate) - ...
    smoothNewInfections(breakthroughDate - 1)) / (breakthroughDate - ...
    1 - vaxDate) * 100;
convergence = (smoothNewInfections(breakthroughDate) - ...
    smoothNewInfections(end)) / (length(newInfections) - ...
    breakthroughDate) * 100;
breakthroughRate = vaxRate - convergence;
```



pre-vaccination phase

```
d = cumulativeDeaths(1);
v = 0; % new parameter: vaccinated
r = 0;
i = newInfections(1);
s = 1 - i - d;
x1 = [s, i, r, v, d]';

initialInfectionRate = mean(newInfections(1:vaxDate - 1));
initialDeathRate = mean(newDeaths(1:vaxDate - 1));

A1 = [1 - initialInfectionRate, 0.08,                        0,  0,  0;
      initialInfectionRate,     0.9,                         0,  0,  0;
      0,                        0.01 - initialDeathRate,     1,  0,  0;
      0,                        0,                           0,  1,  0;
      0,                        initialDeathRate,            0,  0,  1];
```

```
Y1 = x1;
for j = 2:vaxDate - 1
    x1 = A1 * x1;
    Y1 = [Y1, x1];
end
```

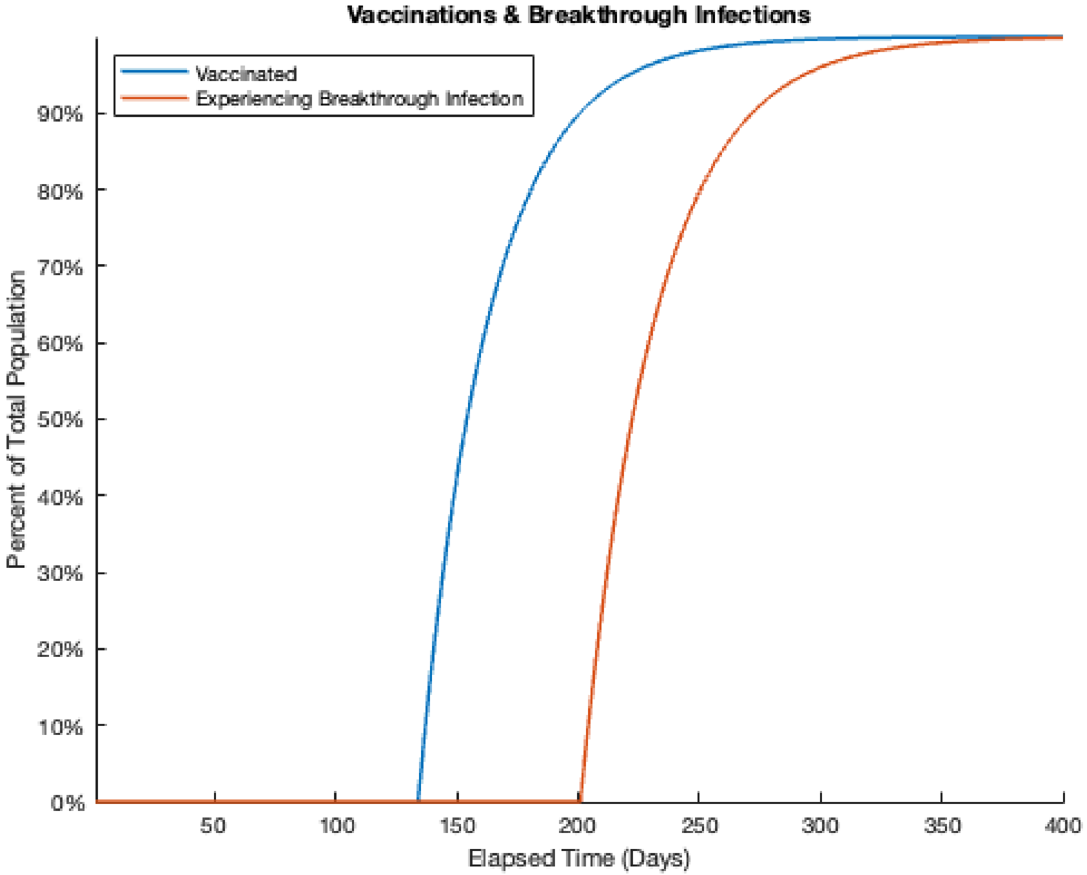
vaccination phase

```
x2 = x1;

vaxInfectionRate = mean(newInfections(vaxDate:length(newInfections)));
vaxDeathRate = mean(newDeaths(vaxDate:length(newDeaths)));

A2 = [1 - vaxInfectionRate - vaxRate,   0.03,                   0,  0,  0;
      vaxInfectionRate,                 0.95,                   0,  0,  0;
      0,                                0.02 - vaxDeathRate,    1,  0,  0;
      vaxRate,                          0,                      0,  1,  0;
      0,                                vaxDeathRate,           0,  0,  1];

Y2 = x2;
for j = vaxDate + 1:length(newInfections)
    x2 = A2 * x2;
    Y2 = [Y2, x2];
end
```

competition

```
vaxpop = zeros(1, vaxDate - 1);
for j = vaxDate:length(newInfections)
    if j == vaxDate
        vaxpop = [vaxpop, vaxRate];
    else
        vaxpop = [vaxpop, vaxpop(j - 1) + vaxRate * (1 - vaxpop(j - 1))];
    end
end
vaxbreak = zeros(1, breakthroughDate - 1);
for j = breakthroughDate:length(newInfections)
    if j == breakthroughDate
        vaxbreak = [vaxbreak, breakthroughRate];
    else
        vaxbreak = [vaxbreak, vaxbreak(j - 1) + breakthroughRate * ...
            (1 - vaxbreak(j - 1))];
    end
end

figure;
hold on;
plot(vaxpop * 100, 'LineWidth', 1.5);
plot(vaxbreak * 100, 'LineWidth', 1.5);
hold off;
axis tight;
title('Vaccinations & Breakthrough Infections');
legend('Vaccinated', 'Experiencing Breakthrough Infection', 'Location', ...
    'northwest');
```

```
xlabel('Elapsed Time (Days)');
ylabel('Percent of Total Population');
ytickformat('percentage');
exportgraphics(gca, 'competition.png');

save('competition.mat', 'vaxpop', 'vaxbreak');
```



Published with MATLAB® R2023a