```
clear, clc;
```

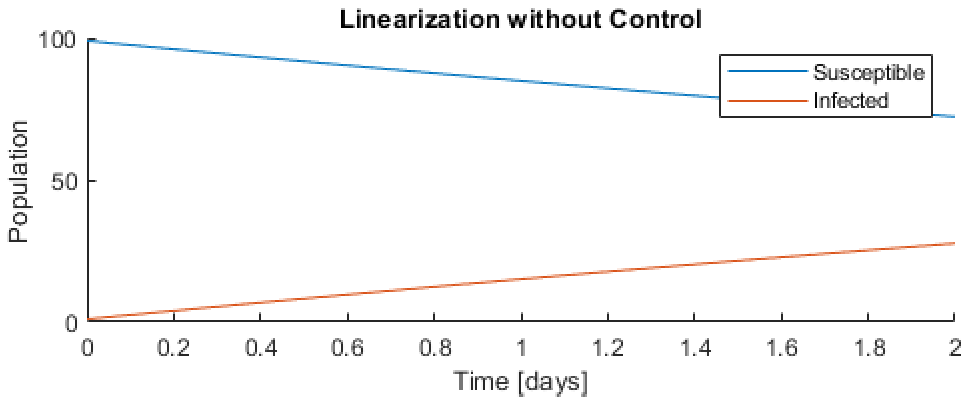
Part 2.3: designing control

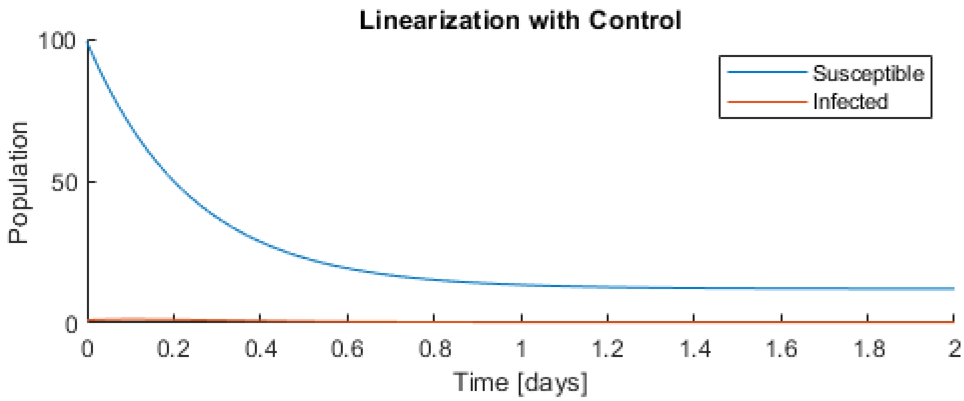
```
%goal is to get large, negative eigenvals.

%Choosing values for K hat just so happened to give negative, large
%eigenvals, for the sake of argument. Maximizing the eigenvals using K is
%not within the scope of the case study, so different values are chosen to
%illustrate the difference

%params
k1 = 5;
k2 = 100;
v1 = 0.9;
x0 = [99 1];
x2_0 = x0(2);
alpha = 0.1;
r = 0.1;

%matricies for statespace K is chosen s.t. the eigenvalues are large and
%negative
K = [4 0; 0 8];
A = Jac(k1,k2,v1,x2_0,alpha,r)-eye(2)*K;
disp("Time constant for matrix with control")
5/min(eig(A))
%Eigenvalues of new system, they are large and both negative,
%time constant = 5/max(eig(A)) is small

B = eye(2);

%input, again chosen because the steady state is large, one could trade off
%a higher steady state for smaller (in magnitude) eigenvalues.
%Also, input is really u(t) = -Kx+u (the u down there )
u = [50; 0];


%gross sim and plotting
[t,x] = ode45(@(t,x) dxx(t,x,A,u,B), [0 2], x0);

%plotting
figure, subplot(2,1,2),hold on

plot(t,x(:,1))
plot(t,x(:,2))
xlabel("Time [days]")
ylabel("Population")
title("Linearization with Control")
legend("Susceptible", "Infected")
hold off

%sim for no control, so A is normal Jacobian around initial condition
```

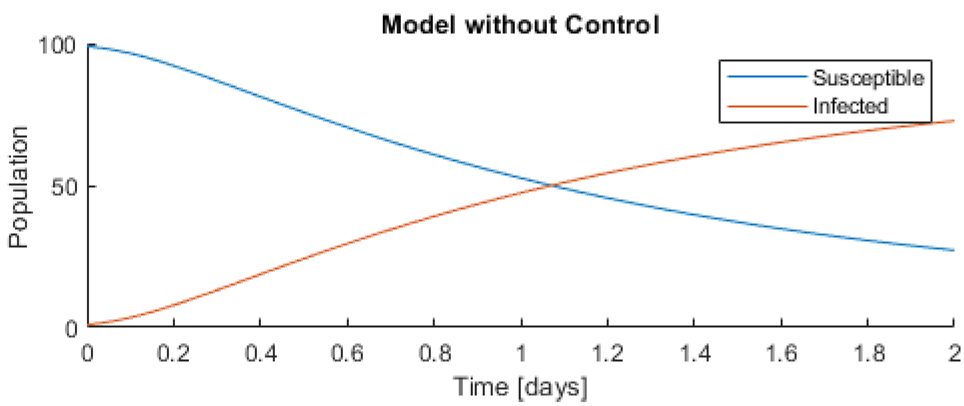
```
A = Jac(k1,k2,v1,x2_0,alpha,r);

disp("Time constant for matrix without control")
5/min(eig(A))

%u = [0;0], only simulating for 2 days cause plots look better and thats
%all the time the control needs to settle :)
[t,x] = ode45(@(t,x) dxx(t,x,A,[0;0],B), [0 2], x0);

%plotting
subplot(2,1,1)
hold on
plot(t,x(:,1))
plot(t,x(:,2))
xlabel("Time [days]")
ylabel("Population")
title("Linearization without Control")
legend("Susceptible", "Infected")
hold off


%applying to non-linear model:

%model with control, same choices as above (actually pretty good, im
impressed)
[t,x] = ode45(@(t,x) nonlin(t,x,k1,k2,v1,alpha,r,K,u), [0 2], x0);

%plotting
figure, subplot(2,1,2),hold on

plot(t,x(:,1))
plot(t,x(:,2))
xlabel("Time [days]")
ylabel("Population")
title("Model with Control")
legend("Susceptible", "Infected")
hold off

%normal dynamics, K = zeros(2) and u^* = [0;0] so [0,0;0,0]*x+ u^* = 0
%forall x and t
[t,x] = ode45(@(t,x) nonlin(t,x,k1,k2,v1,alpha,r,zeros(2),[0;0]), [0 2], x0);

subplot(2,1,1),hold on

plot(t,x(:,1))
plot(t,x(:,2))
xlabel("Time [days]")
ylabel("Population")
title("Model without Control")
legend("Susceptible", "Infected")
hold off
```

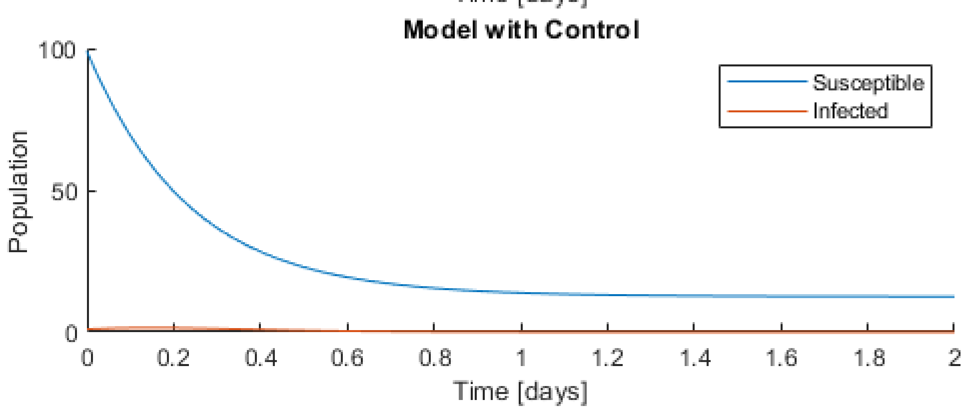
```matlab
function dxdt = nonlin(t,x,k1,k2,v1,alpha,r,K,input)
    u = -K*x + input;
    viral = v1*x(1)*x(2)/(k1+x(2));
    lossImm = alpha*x(2);

    dxdt = [-1*viral + lossImm ;
         viral - lossImm - r*x(2)/(x(2)+k2)] + u;
end

function dxdt = dxx(t,x,A,u,B)

    dxdt = A*x + B*u;

end

function J = Jac(k1,k2,v1,x2_0,alpha,r)
    J = [-v1*x2_0/(k1+x2_0), alpha-k1/(k1+x2_0)^2;
     v1*x2_0/(k1+x2_0), k1/(k1+x2_0)^2-r*k2/(k2+x2_0)^2-alpha];
end
```

```
Time constant for matrix with control

ans =

   -0.6281

Time constant for matrix without control

ans =

  -45.1411
```


Linearization without Control
Population
100
50
0
0 0.2 0.4 0.6 0.8 1 1.2 1.4 1.6 1.8 2
Time [days]
Susceptible
Infected


Linearization with Control
Population
100
50
0
0 0.2 0.4 0.6 0.8 1 1.2 1.4 1.6 1.8 2
Time [days]
Susceptible
Infected


Model without Control
Population
100
50
0
0 0.2 0.4 0.6 0.8 1 1.2 1.4 1.6 1.8 2
Time [days]
Susceptible
Infected


Model with Control
Population
100
50
0
0 0.2 0.4 0.6 0.8 1 1.2 1.4 1.6 1.8 2
Time [days]
Susceptible
Infected

Published with MATLAB® R2024a