
```

close all; clear all;
% Following code is from flow.m

Generating Vector field for Logistic Predator-Prey equation

xlmin = -1; xlmax = 6; x2min = -1; x2max = 6;
xlstep = 0.1; x2step = 0.1;
[x1, x2] = meshgrid(xlmin:xlstep:xlmax, x2min:x2step:x2max);
a = 1.5;
b = 1.1;
c = 2.5;
d = 1.4;
k = 0.5;
dx1 = -a*x1+(b*x1.*x2);
dx2 = c*(1-k*x2).*x2-(d*x1.*x2);
dx1 = dx1./sqrt(dx1.^2 + dx2.^2);
dx2 = dx2./sqrt(dx1.^2 + dx2.^2);
quiver(x1, x2, dx1,dx2,'AutoScaleFactor',0.5)
axis([xlmin xlmax x2min x2max])
% Calling project_system_3_2_2.m for ode45 to find solutions with
  initial
% conditions x1 = 0.5, x2 = 1
%{
function vprime = project_system_3_2_2(t, v)
    x1 = v(1);
    x2 = v(2);
    a = 1.5;
    b = 1.1;
    c = 2.5;
    d = 1.4;
    k = 0.5;

    vprime = zeros(2, 1);

    vprime(1) = -a*x1+(b*x1.*x2);
    vprime(2) = c*(1-k*x2).*x2-(d*x1.*x2);
end
%}
[t_out, v_out] = ode45(@project_system_3_2_2, [0,50], [0.5,1]);

figure(1)
hold on
xlabel('$x_1$', 'Interpreter', 'latex')
ylabel('$x_2$', 'Interpreter', 'latex')
title('Vector field example', 'Interpreter', 'latex')
x2_=@(x) (1/k)-(d/(c*k))*x;
x=-1:0.1:6;
n0=plot(x, x2_(x)); n0.LineWidth = 1; n0.Color = 'k';

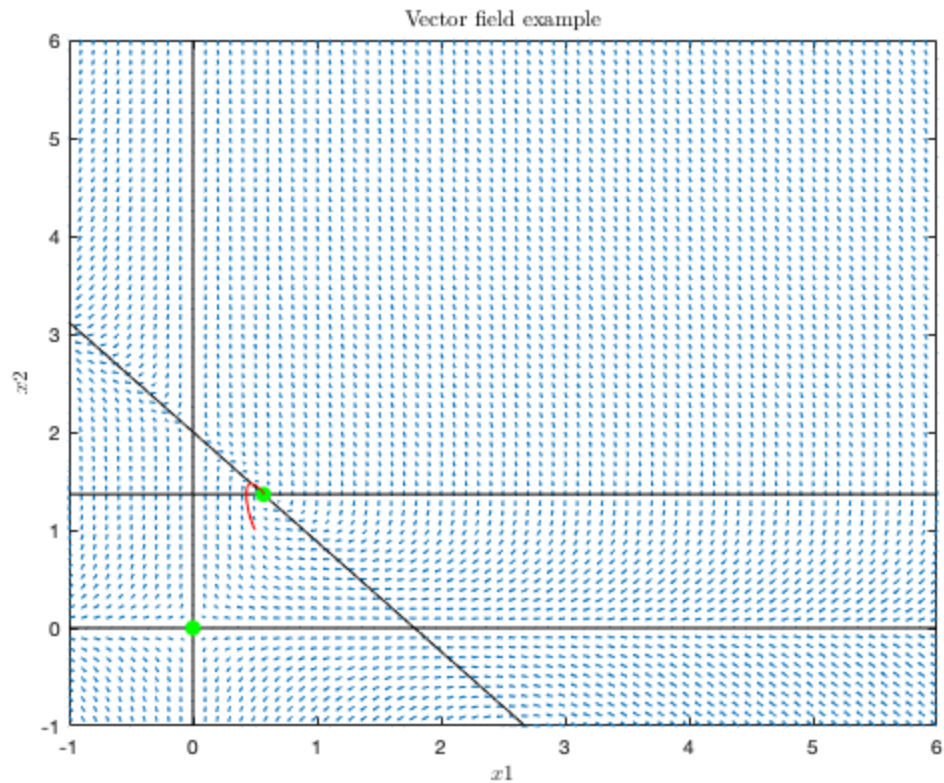
% Adding nullclines and equilibrium solutions
f=a/b;
n3 = line([0 0], ylim); n1 = reffline([0 a/b]);

```

```

n2 = reffline([0 0]);
n1.LineWidth = 1; n2.LineWidth = 1; n3.LineWidth = 1;
n1.Color = 'k'; n2.Color = 'k'; n3.Color = 'k';
plot((c*b-a*c*k)/(b*d),a/b, 'g.', 'MarkerSize', 20);
plot(0,0, 'g.', 'MarkerSize', 20);
plot(v_out(:,1), v_out(:,2),'LineWidth',1,"Color",'r')
hold off

```

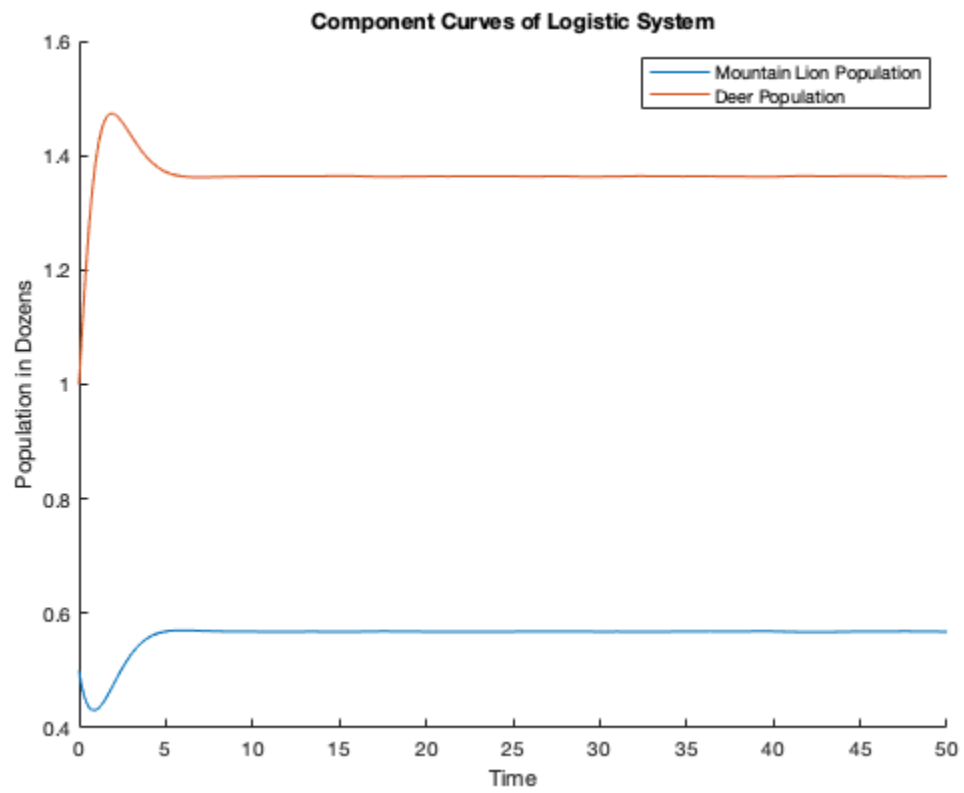
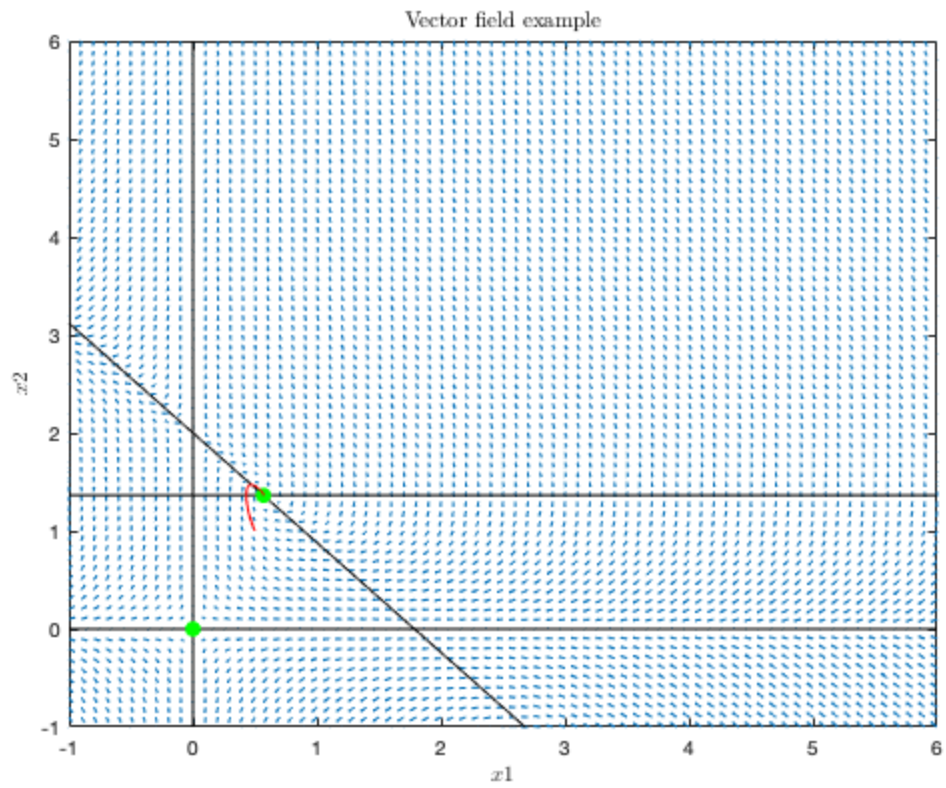


Component Curves of Logistic Predator-Prey equation

```

figure(2)
hold on
plot(t_out, v_out(:,1))
plot(t_out, v_out(:,2))
legend('Mountain Lion Population','Deer Population')
title('Component Curves of Logistic System')
xlabel('Time')
ylabel('Population in Dozens')
hold off

```



Published with MATLAB® R2018b