```
close all; clear all;
% The following code is from flow.m
```

Generating vector field for Lotka-Volterra System of equations

```
x1min = -1; x1max = 6; x2min = -1; x2max = 6;
x1step = .2; x2step = .2;
[x1, x2] = meshgrid(x1min:x1step:x1max, x2min:x2step:x2max);
a = 1.5; b = 1.1; c = 2.5; d = 1.4;
dx1 = -a*x1+b*x1.*x2;
dx2 =  c*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([x1min x1max x2min x2max])

% Calling project_system_3_1_5.m for ode45 to find solutions with
 initial
% conditions x1=0.5 and x2=1
%{
function vprime = project_system_3_1_5(t, v)
    x1 = v(1);
    x2 = v(2);
    a = 1.5;
    b = 1.1;
    c = 2.5;
    d = 1.4;

    vprime = zeros(2, 1);

    vprime(1) = -a*x1+(b*x1.*x2);
    vprime(2) = c*x2-(d*x1.*x2);
end
%}
[t_out, v_out] = ode45(@project_system_3_1_5, [0,20], [0.5,1]);
figure(1)
hold on
    xlabel('$x1$','Interpreter','latex')
    ylabel('$x2$','Interpreter','latex')
    title('Vector field for system','Interpreter','latex')

plot(c/d,a/b, 'g.', 'MarkerSize', 20);
plot(0,0, 'g.', 'MarkerSize', 20);

% Adding nullclines and equilibirum points
e = c/d;
n1 = refline([0 a/b]); n2 = refline([0 0]);
n1.Color = 'k'; n2.Color = 'k';
n1.LineWidth = 1; n2.LineWidth = 1;
n3 = line([0 0], ylim); n4 = line([e,e],[-1,6]);

n3.LineWidth = 1; n4.LineWidth = 1;
```
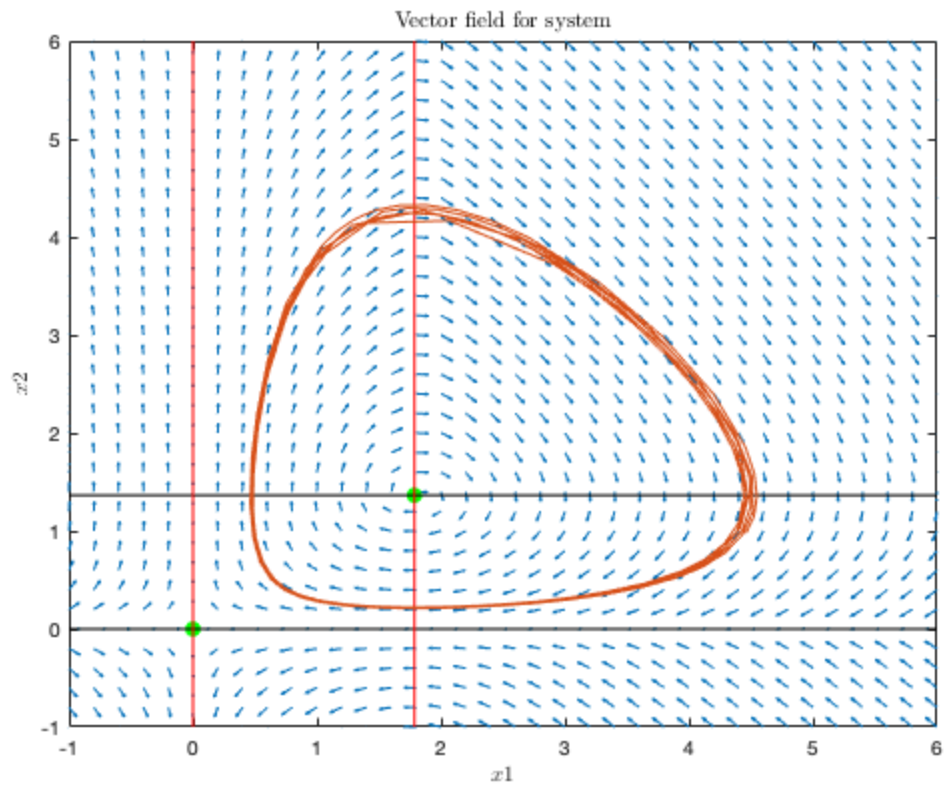
```
n3.Color = 'r'; n4.Color = 'r';
plot(v_out(:,1), v_out(:,2))
hold off
```
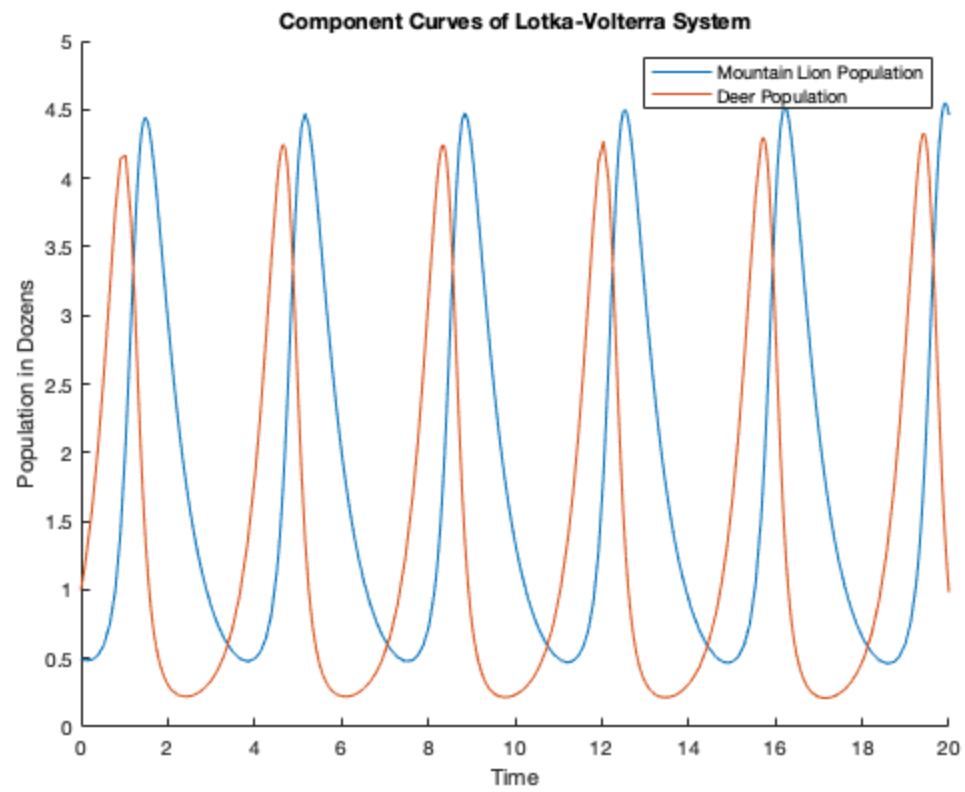


Vector field for system

Component Curves of Lotka-Volterra System

```
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 Lotka-Volterra System')
xlabel('Time')
ylabel('Population in Dozens')
hold off
```

**Component Curves of Lotka-Volterra System**

*Published with MATLAB® R2018b*