Table of Contents


```
%Mathias Frazier
%Section 242
%Matlab Project 1
```

Task 1

```
A = [ 1 2 -3; 0 4 4; -2 5 1];
B = [-4 0 1; 2 2 -2; 7 -5 2];

A*B
B*A


ans =

   -21    19    -9
    36   -12     0
    25     5   -10


ans =

    -6    -3    13
     6     2     0
     3     4   -39
```

Task 2

```
C2 = sym([2 -5; 1 -2]);
[evect,eval] = eig(C2)


evect =

[2 - 1i, 2 + 1i]
[     1,      1]
```

```
eval =

[-1i,   0]
[  0, 1i]
```

Task 3

```
C3 = sym([-1 -1 0; 5 -3 0; 0 0 2]);
[evect,eval] = eig(C3)
```

```
evect =

[0, 1/5 - 2i/5, 1/5 + 2i/5]
[0,          1,          1]
[1,          0,          0]


eval =

[2,        0,        0]
[0, - 2 - 2i,        0]
[0,        0, - 2 + 2i]
```

Task 4

```
syms x(t) y(t);
[xsoln,ysoln] = dsolve(diff(x) == 3*x - y, diff(y) ==  4*x - 2*y)
```

```
xsoln =

C1*exp(2*t) + (C2*exp(-t))/4


ysoln =

C1*exp(2*t) + C2*exp(-t)
```

Task 5

```
[xsoln,ysoln] = dsolve(diff(x) == 3*x - y, diff(y) ==  4*x - 2*y, x(0)==2,
y(0)==-3)
```

```
xsoln =
```

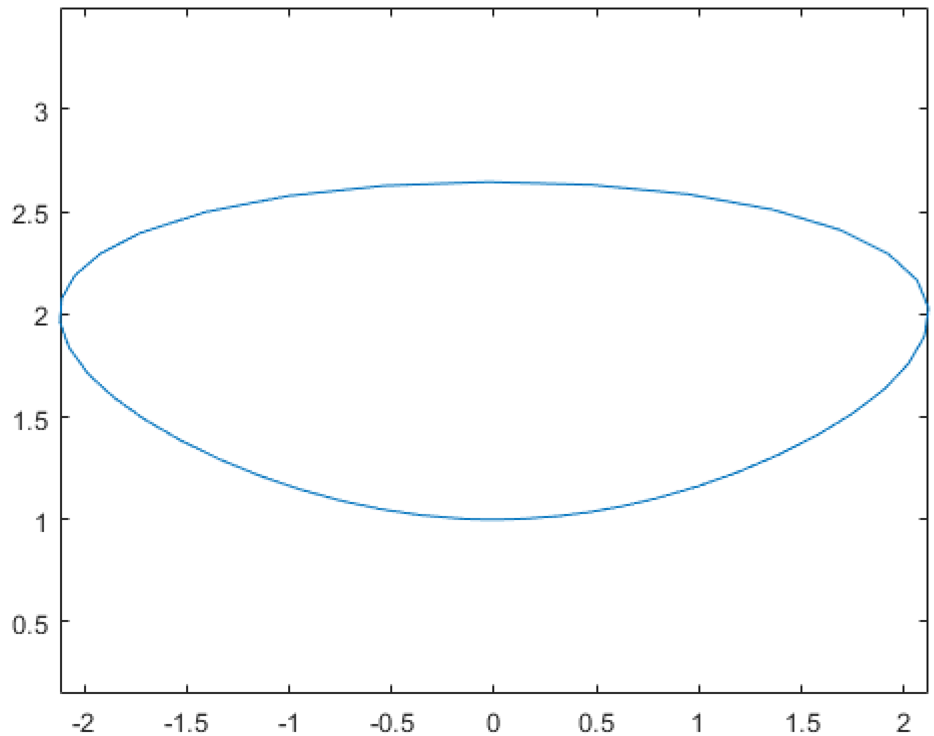
```
(11*exp(2*t))/3 - (5*exp(-t))/3


ysoln =

(11*exp(2*t))/3 - (20*exp(-t))/3
```

Task 6

```
X=[x;y];
A = [2 -5; -1 2];
[xsoln,ysoln] = dsolve(diff(X) == A*X)


xsoln =

5^(1/2)*C2*exp(-t*(5^(1/2) - 2)) - 5^(1/2)*C1*exp(t*(5^(1/2) + 2))


ysoln =

C1*exp(t*(5^(1/2) + 2)) + C2*exp(-t*(5^(1/2) - 2))
```

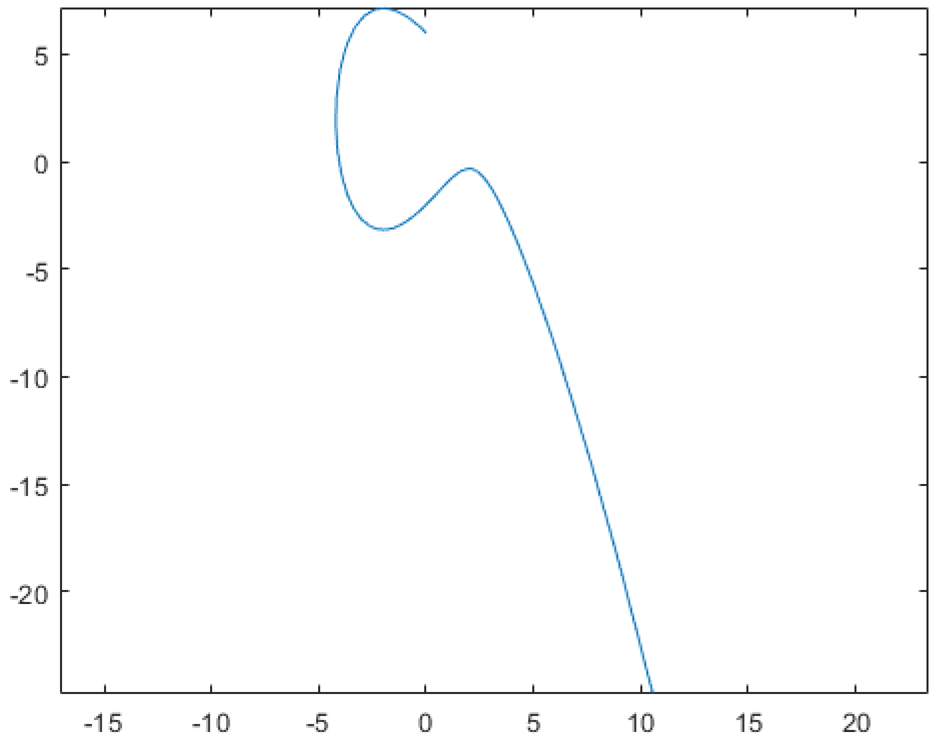
Task 7

```
f = @(t,x) [4*x(2)-x(2)^3;x(1)];
[t,xsoln] = ode45(f,[0 0.821*pi],[0,1]);
figure(7), plot(xsoln(:,1),xsoln(:,2))
axis equal
% rpi is roughly 0.82pi, so r = 0.82
```



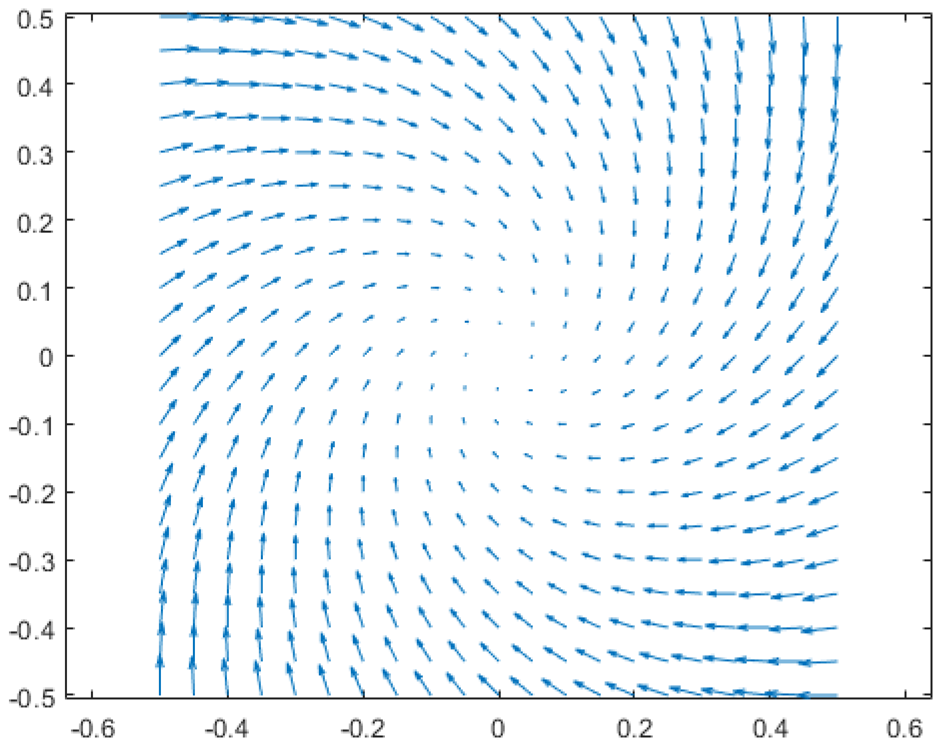
Task 8

```
f = @(t,x) [2 - x(2);4 - x(1)^2];
[t,xsoln] = ode45(f,[0 4],[0,6]);
figure(8), plot(xsoln(:,1),xsoln(:,2))
axis equal
```



Task 9

```
[x,y] = meshgrid(-0.5:0.05:0.5, -0.5:0.05:0.5);
xprime = -x - 2*x^2.*y + y;
yprime = -x-y;
figure(3);
quiver(x,y,xprime,yprime)
axis equal

% Based off of the phase portrait, the stationary solution of x(t)=0 and
% y(t)=0 is stable, as all the arrows, or starting points point towards 0.
```



Published with MATLAB® R2024a