Drexel University

Office of the Dean of the College of Engineering

ENGR 232 – Dynamic Engineering Systems

Recitation Guidelines for Week 2

Topics for recitation this week:

- Worked example on separation of variables and integrating factor method
- Linear algebra for solving for the equilibrium points
- Introduction to numerical methods, ODE45 and MSE (mean square error) computation

---

1. Separation of Variables

Recall that first-order differential equations are **separable** if they have the form:

$$\frac{dy}{dt} = f(y)g(t)$$

Furthermore, all first-order **autonomous** differential equations are separable with $g(t) = 1$, since autonomous DEs don't depend on the independent variable.

Examples:

a.

$$\frac{dy}{dt} = \ln(ty)$$

This is **not** separable.

---

b.

$$\frac{dy}{dt} = t\ln(y^{2t}) + t^2$$

This is separable as we can see after a little manipulation:

$$\frac{dy}{dt} = 2t^2\ln(y) + t^2 = t^2 \cdot (2\ln(y) + 1)$$

Log of a Power

$\ln(x^p) = p\ln(x)$

---

c.

$$\frac{dy}{dt} = \frac{1 + 2t}{4 + 2y}$$

This is separable! To find the solutions, just separate and integrate!

$$\int (4 + 2y)\,dy = \int (1 + 2t)dt \quad \rightarrow \quad 4y + y^2 = t + t^2 + c$$

Implicit form

To find the solution in explicit form, just add 4 to each side and complete the square (on the LHS).

$$(y + 2)^2 = t + t^2 + c + 4$$

Thus:

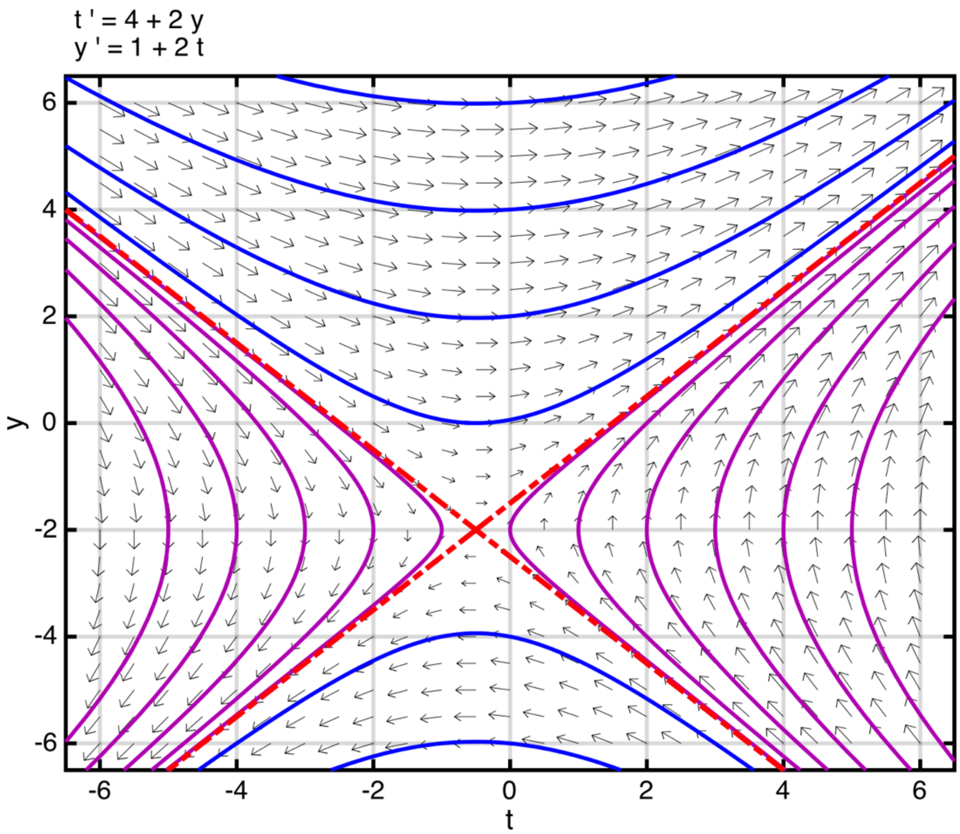
$$y = \pm\sqrt{t + t^2 + k} - 2$$

Explicit form

We defined $k = c + 4$

Here's a plot of selected solutions curves that was found using **pplane8**.

Can you find the equations for the two straight lines that can be seen on the graph?



Example: Compound Interest – Watch your money grow!

Consider a compound interest model where a retired person has a sum $S(t)$ invested so as to draw interest at an annual rate r compounded continuously. This person also withdraws money for living expenses at k dollars per year. Assuming withdrawals are made continuously, we can model this by the differential equation:

$$\frac{dS}{dt} = rS - k$$

To solve, just separate and integrate!

$$\frac{1}{rS - k} dS = 1\, dt$$

Separate the variables

$$\int \frac{1}{rS - k} dS = \int 1\, dt$$

Integrate

$$\frac{1}{r}\ln(rS - k) = t + C$$

Assuming $rS > k$

$$rS - k = Ae^{rt}, \quad A = e^{rC}$$

Exponentiate both sides, then use law of exponents.

$$S = \frac{A}{r} e^{rt} + \frac{k}{r}$$

Using the initial condition:

$$S(0) = S_0$$

The initial investment was S_0.

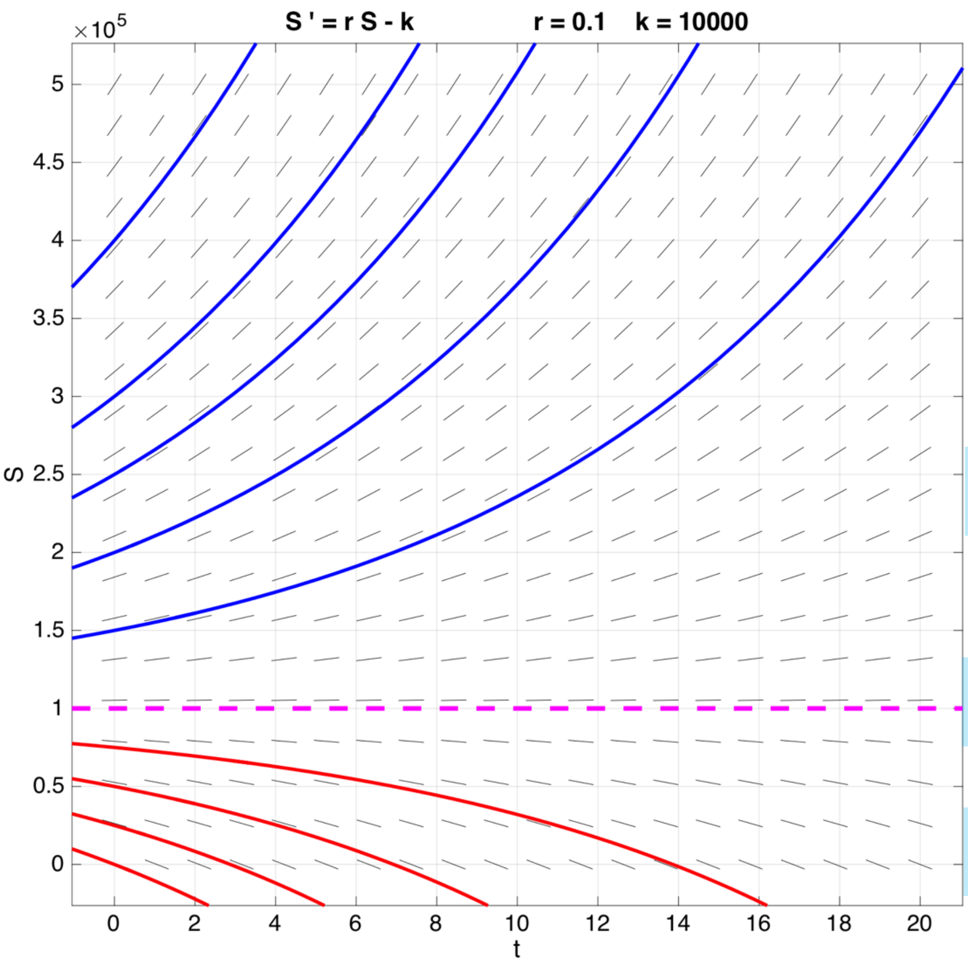
$$\frac{A}{r} = S_0 - \frac{k}{r}$$

Thus, we get:

$$\boxed{S = \left(S_0 - \frac{k}{r}\right) e^{rt} + \frac{k}{r}}$$

As long as the initial investment satisfies $rS(0) > k$, this saver's wealth will continue to grow exponentially despite the withdrawals!

In the figure below, we let r = 10% and withdraw k = \$10,000 every year. Notice the scale marker $\times 10^5$, so the amounts are multiples of \$100,000. The dotted line shows that the balance remains constant for an initial investment of \$100,000. Larger initial investments continue to build faster than the yearly withdrawal. (**blue** curves.). But smaller initial investments, (shown in **red**) are not enough, and the saver eventually runs out of money.



Savings continue to grow, despite yearly withdrawals.

The initial investment of \$100,000 remains constant.

The savings will eventually runout!

2. Integrating Factor Method for 1st-Order Linear Differential Equations

A first-order **linear** differential equation can be expressed in the following ***standard*** form:

$$\frac{dy}{dt} + p(t)y = q(t)$$

You may have to rearrange the equation to get it into this form. From here, we define the **integrating factor** as:

$$\mu(t) = e^{\int p(t)dt}$$

In lecture, it is shown that the solution to this differential equation is obtained from:

$$\mu(t)\, y(t) = \int \mu(t)q(t)\, dt + C$$

Expression for the solution, using the integrating factor.

Example:
Consider again the compound interest model:

$$\frac{dS}{dt} = rS - k$$

This is not in the standard form, so we rearrange as follows:

$$\frac{dS}{dt} - rS = -k$$

Rearranged to standard form.

Comparing this to the general form, we see that $p(t) = -r$ and that $q(t) = -k$

The values of p and q.

From here, we can compute the **integrating factor** as:

$$\mu(t) = exp\left(\int -r\, dt\right) = e^{-rt}$$

Thus, the solution can be obtained from substituting into the equation above, giving:

$$e^{-rt}\, S = \int (-k)e^{-rt}dt + C$$

$$e^{-rt}S = \frac{k}{r}e^{-rt} + C$$

The initial condition $S(0) = S_0$, allows us to find the constant of integration:

$$C = S_0 - \frac{k}{r}$$

Thus, the solution matching the initial investment is:

$$S = \left(S_0 - \frac{k}{r}\right)e^{rt} + \frac{k}{r}$$

This is the same result we obtained previously.

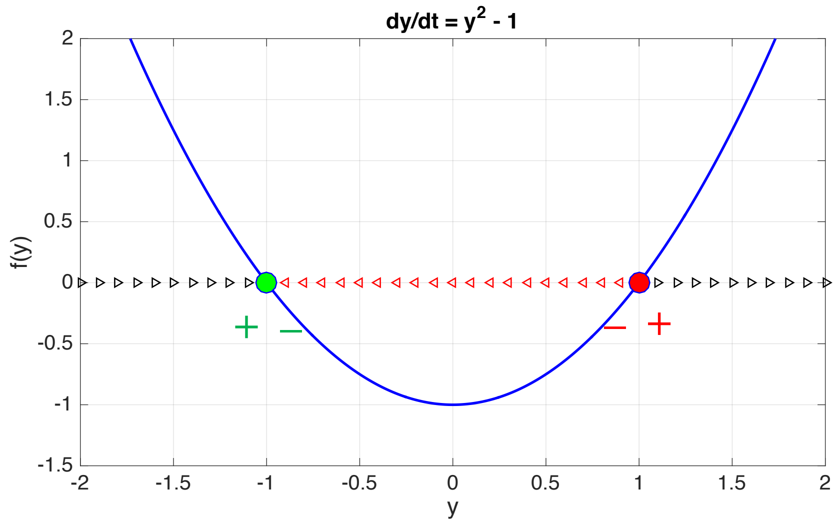
Note: All linear first-order differential equations can be solved using this technique, provided you can find the integrals.

Section B: Linear algebra for solving for equilibrium points

1. Find the equilibrium points for the given differential equation, identify their stability, and give the region of attraction (ROA) for each.

a. $y' = y^2 - 1$

To find the equilibrium points, just set the derivative to zero, and solve for the roots.



Equilibrium points: ***y* = –1, 1**

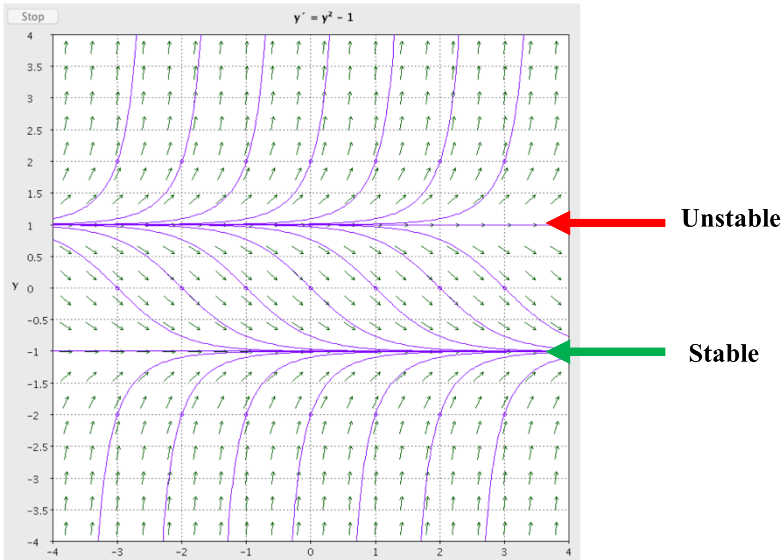
The critical point $y = -1$ is **stable**. That's shown by the **green** marker. Points slightly above or below it, converge back to it. It's **region of attraction** consists of all the points that converge to it. See the arrows on the above graph.

ROA(–1): $-\infty < y < 1$ or in interval notation: ROA(–1) = $(-\infty, 1)$

The critical point $y = +1$ is **unstable**. That's shown by the **red** marker. Points slightly above or below it, diverge away! It's **region of attraction** is just itself.

ROA(+1) = [1, 1]

You can also see the stability or instability by plotting the **direction field**. We see the solution curve $y = +1$ is unstable, because nearby curves diverge away from it. Whereas $y = -1$ is stable, because nearby curves converge towards it. Also notice that as for all **autonomous** systems, all solution curves in a horizontal strip between the critical points can be obtained by shifting just one **horizontally**.

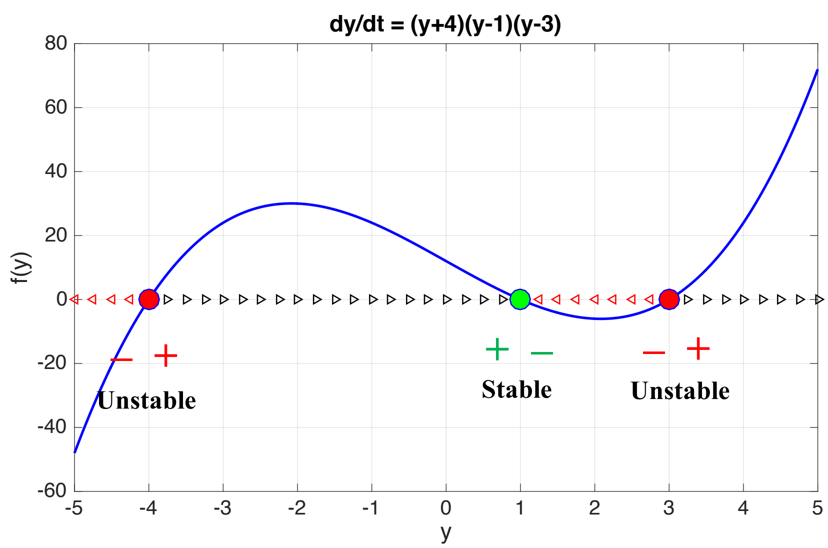


---

b. Find the equilibrium points for this next differential equation, identify their stability, and give the region of attraction (ROA) for each.

$$\frac{dy}{dt} = (y+4)(y-1)(y-3)$$

To find the equilibrium points, just set the derivative to zero, and solve for the roots.



Equilibrium points: $y = -4, 1, 3$

***y* = −4 unstable**

ROA(−4) = [−4, −4]

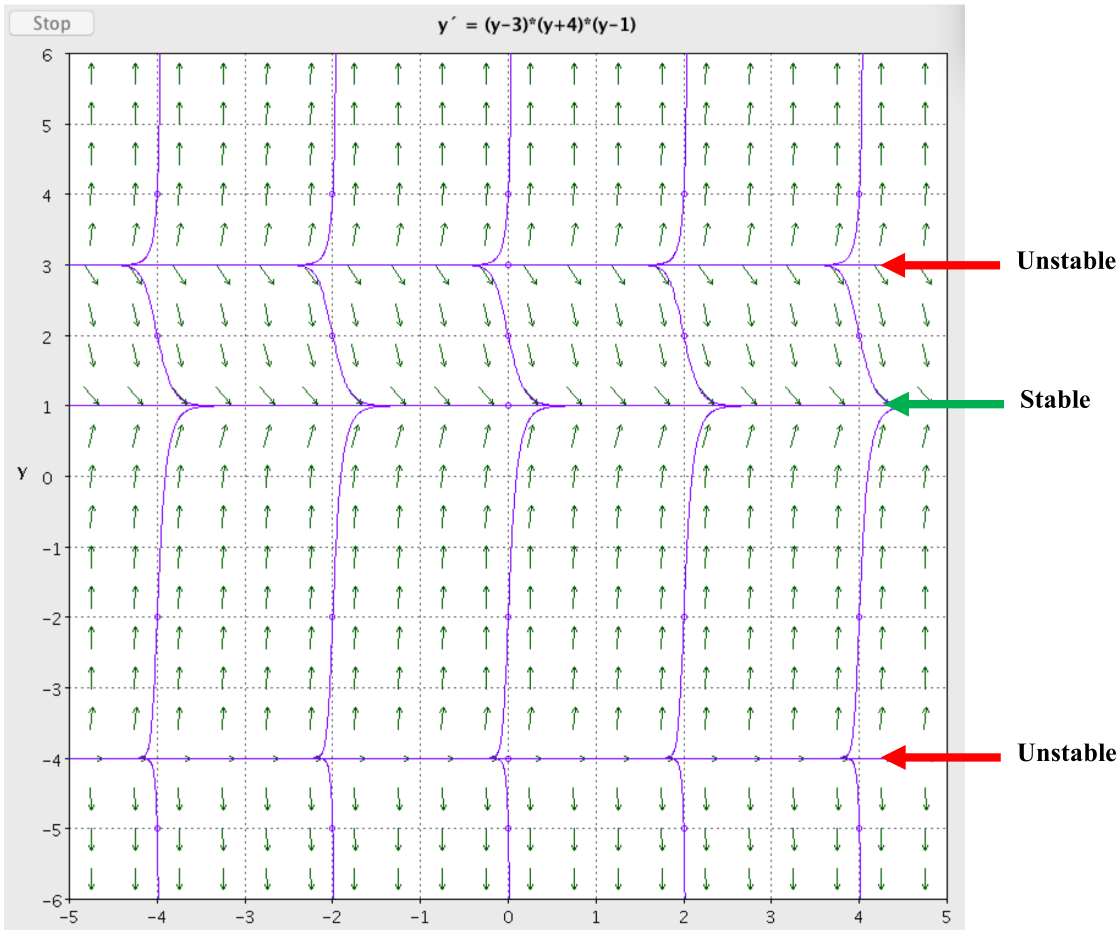
***y* = +1 stable**

ROA(+1) = (−4, 3)

***y* = +3 unstable**

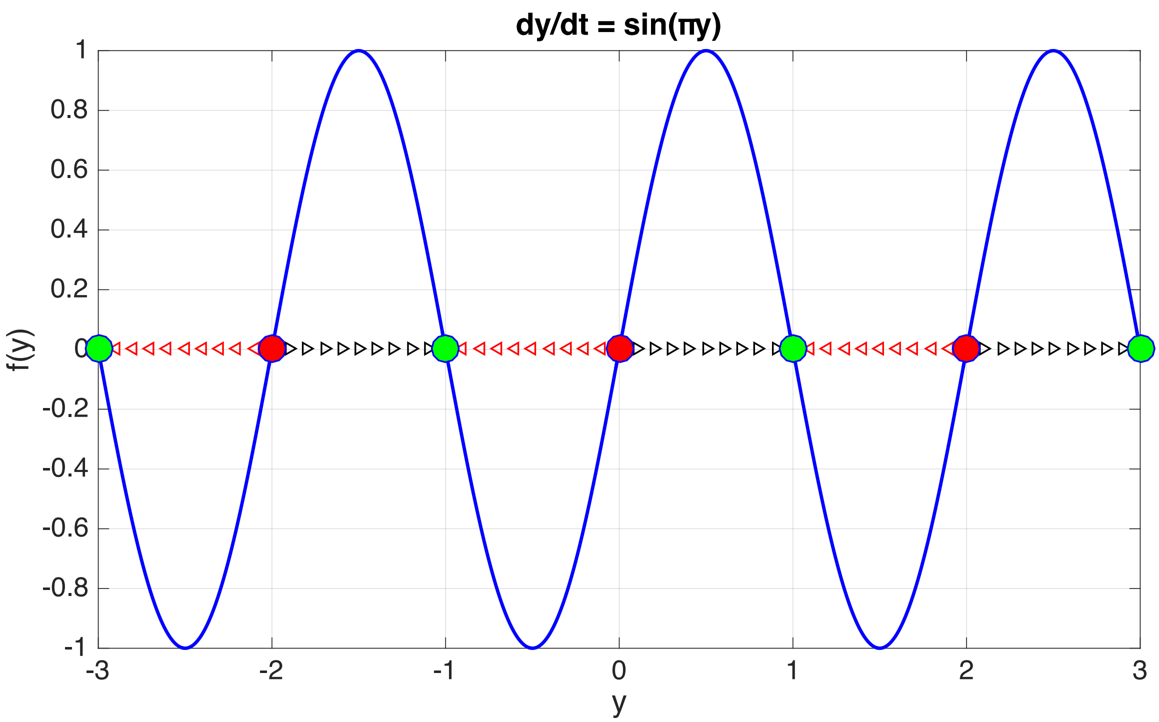
ROA(+3) = [3,3] Just itself.

You can also see the stability or instability by plotting the direction field. We see the solution curves $y = +3$ and $y = -4$ are unstable, because nearby curves diverge away from them. Whereas $y = +1$ is stable, because nearby curves converge towards it. Also notice that as for all **autonomous** systems, all solution curves in a region can be obtained by shifting just one **horizontally**.



c. Find the equilibrium points for this next differential equation, identify their stability, and give the region of attraction (ROA) for each.

$$y' = \sin \pi y$$



Equilibrium points at $y = n$ where n is an integer. That's an infinite number of equilibrium points!

For even $k = 2n$, **unstable**. See **red** dots. **ROA**($2n$) = [$2n$, $2n$] Just itself.

For odd $k = 2n + 1$, **stable**. See **green** dots. **ROA**($2n$+1) = ($2n$, $2n$+2)

Stability could also easily be checked by plotting solutions using the direction field.

Section C: Numerical Methods

For first-order system models, if the equation is separable we can find a solution through separation of variables, providing that the integrals in the separated form exist. For linear first-order system models, we also have the added benefit of being able to solve using the integrating factor method.

Not all equations will have a closed-form solution, and at times, the integrals required to solve systems may be difficult to compute. Furthermore, as shown in the lecture notes, it is not always easy (or possible) to express the solution in the **explicit** form: y=$f(t)$. Sometimes, we have to settle for an **implicit** solution: $F(t, y) = c$

This example is directly from the pre-lab.

Example 1: Consider the 1st-order, separable, non-linear, non-autonomous differential equation:

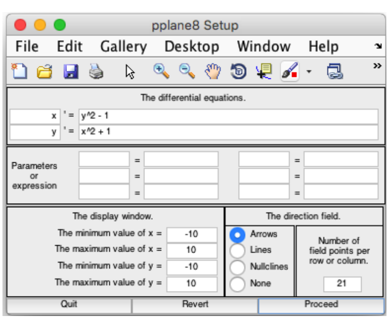
$$\frac{dy}{dx} = \frac{x^2+1}{y^2-1}$$

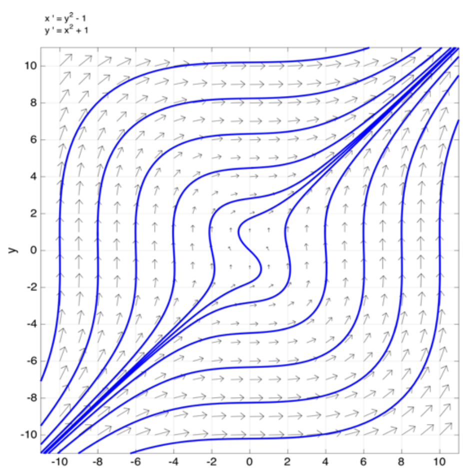
This has an implicit solution via separation of variables: $y^3 - 3y - (x^3 + 3x + 3c) = 0$

Let's show that. Just separate and integrate!

$(y^2 - 1)\, dy = (x^2 + 1)\, dx$	Separate the variables.
$\int (y^2 - 1)\, dy = \int (x^2 + 1)\, dx$	Prepare to integrate both sides.
$\frac{y^3}{3} - y = \frac{x^3}{3} + x + c$	Integrate, and add the constant of integration.
$y^3 - 3y - (x^3 + 3x + 3c) = 0$	Rearrange.

This is not the most straightforward equation to plot, since you need to solve the equation for $y(t)$ unless you have access to an implicit plotting routine. In these situations, a numerical solution such as shown below using **pplane8** will suffice to analyze the system behavior.





Challenge: See if you can create these solution curves using pplane8

Example 2: We will see an implementation of these methods using the following 1st-order, non-linear, autonomous example:

$$\frac{dy}{dt} = f(t,y) = 4y - y^2, \quad y(0) = 1$$

a. Euler's Method – (See lecture notes from week 1)

Euler's method, as we derived in the class notes last week, requires us to first write a function that gives the derivative. Start by defining an anonymous function in MATLAB as follows:

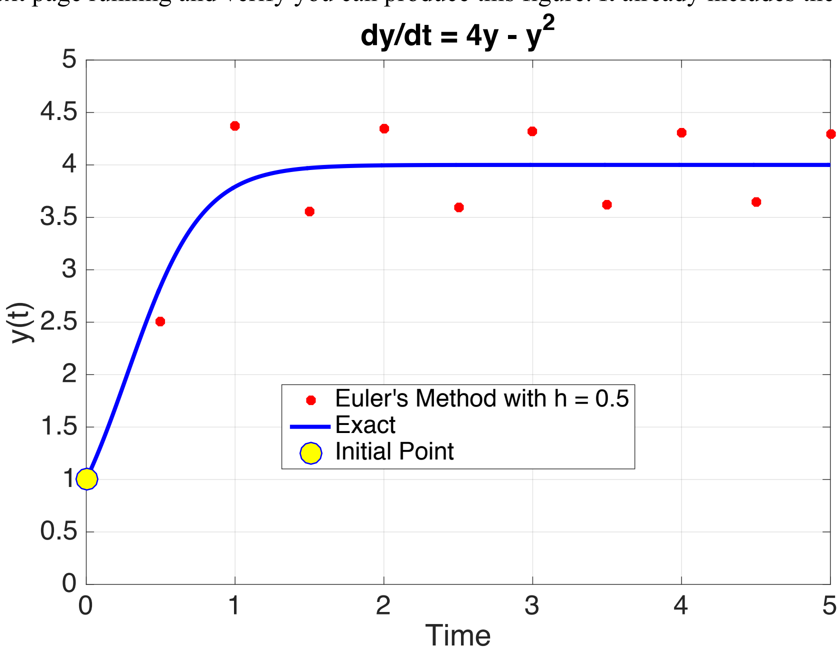
```
f = @(t,y)  4*y - y^2  % Anonymous function for the slope dy/dt.
```

Inputs **Output**

Learn more about anonymous functions here: **http://www.mathworks.com/help/matlab/matlab_prog/anonymous-functions.html**

Note: The function has to be written with inputs (t, y) in the correct order. This is needed later for **ode45** so we will adopt this convention for defining all first order functions.

Get the code on the next page running and verify you can produce this figure. It already includes the above function.



A smaller step size h, will give a more accurate solution.

There's just lots of great code on the next page. You should be able to reproduce almost all of it before the MATLAB Final in the last week of classes. For now, it is OK to just get the general idea.

As shown in the class notes, Euler's method solves the differential equation at a series of time points. We have to know the start time, the end time for the computation, the initial value of y and the step size. A smaller step size gives a more accurate solution, though it takes longer to compute the solutions.

Code for the exact solution using **dsolve** and a numerical solution using **Euler's Method**.

```
%% Pre-Lab 2 Fall 2016
% ENGR 232 Week 2 Euler's Method Example and exact solution using dsolve

clear, clc, close all
% First declare the function f(t,y) which gives the slope dy/dt.
f = @(t,y)  4*y - y^2  % Anonymous function.

% Initialize the Variables
dt = 0.5;  % Step size. Also called h in the notes.
tStart = 0;    yStart = 1; % Initial point.
tEnd = 5;          % Stopping time.

% Define time points and solution vector
t_points = tStart: dt: tEnd;
y_points = zeros(size(t_points));  % Use zeros as place holders for now.

% Initialize the solution at the initial condition.
y_points(1) = yStart;

% Implement Euler's method using a for loop.
for i=2:length(t_points)
    yprime = f(t_points(i-1),y_points(i-1));
    y_points(i) = y_points(i-1) + dt*yprime;
end

% Plot Solutions
figure(1)
plot(t_points, y_points, 'red*', 'LineWidth', 3)
grid on
xlabel('Time')
ylabel('y(t)')
set(gca, 'FontSize', 20)
title('dy/dt = 4y - y^2')
axis([0 5 0 5])
hold on

% Now let's find the exact solution.
syms y(t)
DE = diff(y,t) == 4*y - y^2;
sol = dsolve(DE, y(0) == 1)
Y = matlabFunction(sol)
t_points = tStart: 0.01: tEnd;
plot(t_points, Y(t_points), 'blue', 'LineWidth', 3)
plot(tStart, yStart, 'blueo', 'MarkerSize', 16, 'MarkerFaceColor', 'yellow')

legend('Euler''s Method with h = 0.5', 'Exact', 'Initial Point', 'Location', 'Best')
```

Rerun your code, but change the step-size for Euler's method to the smaller value h = 0.1 (called dt in the code) instead of 0.5, which was really too large. Create this new image. Be sure to adjust the legend to state the new value for h.