Euler's Method and Fourth Order Runge-Kutta

In the introduction lecture I mentioned how slope fields were related to numerical methods for estimating solutions to differential equations. In this lecture I will develop the algebra for Euler's method and briefly indicate how a method used in industry, Runge-Kutta, is related to Euler's method. This may entice you to take a Numerical Analysis class where the theory of numerical methods is treated in detail.

Let us return to the example we saw in the introduction: $y' = y^2 - x$, $y(0) = 0$. The slope field is below. Euler's method generates a set of points; attaching these points with successive line segments approximates the graph of the solution. Below is the process used to generate the set of points. I will use a step size of $h = 1/2$. The smaller the step size, the more accurate the method; however, if the step size is too small, there will be machine round-off error.

$f(x,y) = y^2 - x$

Begin with (x_n, y_n)

$\boxed{y_{n+1} = y_n + h f(x_n, y_n)}$

$y_1 = 0 + \frac{1}{2}(0-0) = 0$

$y_2 = 0 + \frac{1}{2}\left(0 - \frac{1}{2}\right) = -\frac{1}{4}$

$y_{-1} = 0 - \frac{1}{2}(0) = 0$

$y_{-2} = 0 - \frac{1}{2}\left(0^2 - (-.5)\right)$

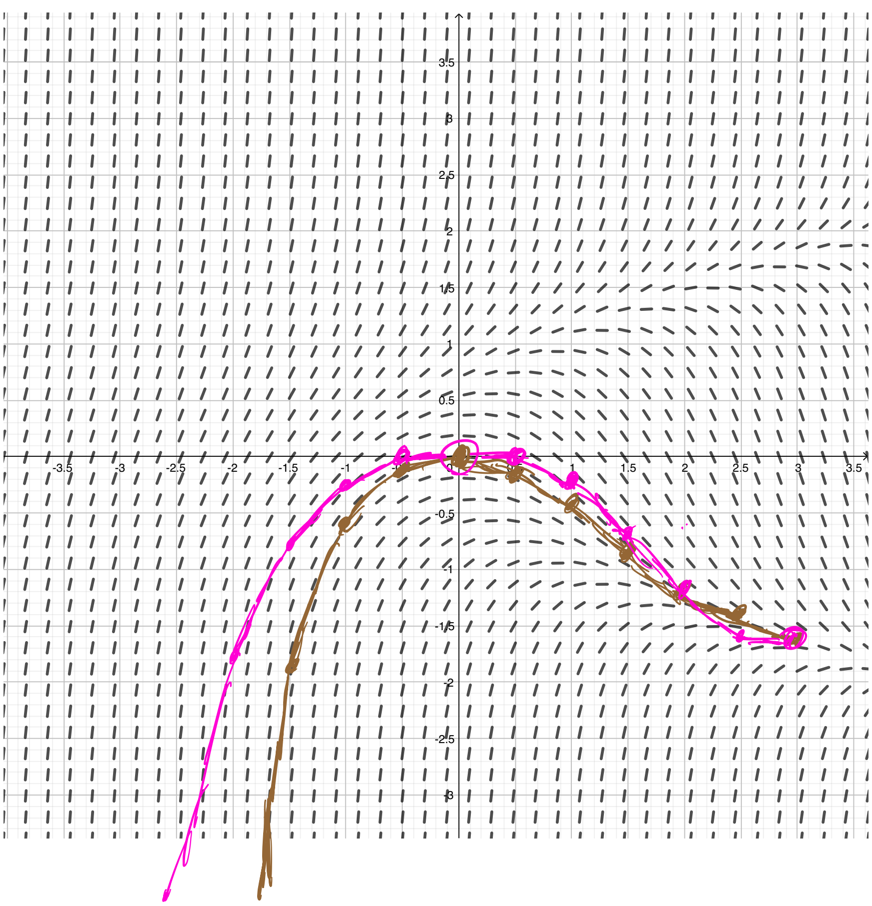
$y_3 = y_2 + \frac{1}{2}(y_2^2 - x_2) = (-.25) + \frac{1}{2}\left(\frac{1}{16} - 1\right)$

Here is the rest of the table: Euler

n	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
x_n	-3	-2.5	-2	-1.5	-1	-0.5	0	0.5	1	1.5	2	2.5	3
y_n	-16	-4.52	-1.84	-0.78	-.25	0	0	0	-.25	-0.72	-1.21	-1.64	-1.64

(copy error?)

Now let's graph the points on the slope field for $y' = y^2 - x$.



RK-4

n	x_n	y_n
-4	-2	-20.43
-3	-1.5	-1.79
-2	-1	-0.56
-1	-0.5	-0.12
0	0	0
1	0.5	-0.12
2	1	-0.46
3	1.5	-0.86
4	2	-1.19
5	2.5	-1.44
6	3	-1.63

Graph this estimate on th

Runge-Kutta works the same way except it calculates the slope differently: it uses a weighted average of four slopes. The weighted average uses the slopes

$$m_1 = f(x_n, y_n)$$

$$m_2 = f\left(x_n + \frac{h}{2}, y_n + \frac{hm_1}{2}\right)$$

$$m_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{hm_2}{2}\right)$$

$$m_4 = f(x_n + h, y_n + hm_3)$$

Then

$$y_{n+1} = y_n + \frac{h}{6}(m_1 + 2m_2 + 2m_3 + m_4)$$

Notice how the slope is found using a weighted average similar to Simpson's rule that was used in second semester calculus to estimate integrals. The error for Runge-Kutta is proportional to h^4 whereas the error for Euler's method is proportional to h, and so greater accuracy can be obtained before exhausting the machine's capability.

MatLab uses the **ode45** command to produce an estimate using Runge-Kutta. You will use this in the CAS problem in homework #2. MatLab has no such command for Euler's method (because error is too large for it to be useful) so you will need to create your own function to do that. I have written the function for you in the MatLab notes in homework #2.

Here is a table of points generated by Runge-Kutta for $y' = y^2 - x$, $y(0) = 0$.

n	x_n	y_n
-4	-2	-20.43
-3	-1.5	-1.79
-2	-1	-0.56
-1	-0.5	-0.12
0	0	0
1	0.5	-0.12
2	1	-0.46
3	1.5	-0.86
4	2	-1.19
5	2.5	-1.44
6	3	-1.63

Graph this estimate on the original slope field and compare with the Euler method estimate.