Lesson 3: Numerical methods

William McCallum

30 August 2005

Review

Last time we used a computer to draw some slope fields and solutions to differential equations. Let's think a little bit about what is really going on.

Look at

$$x' = f(t, x)$$

as a set of instructions for how to construct the function x. You want to draw a graph whose slope at every point $(t, x(t))$ on its graph is $f(t, x(t))$. For example, if $f(t, x) = 0.1x$, then we want the slope to be $0.1x(t)$. Of course, in this case we know that

$$x(t) = e^{0.1t}$$

has slope

$$x'(t) = 0.1e^{0.1t} = 0.1x(t)$$

at every point on its graph. So it satisfies the d.e.

$$x' = 0.1x.$$

Slope fields

In the last example we had to guess the solution. We can't find it by integrating both sides. We already knew the properties of $e^{0.1x}$ and could write a d.e. describing those properties. But we want to go the other way; given a d.e., find the solution.

If we can't guess the soluion, we can visualize its graph. We have to imagine drawing a graph that has the right slope at every point. One way to do this, slope fields, is to plaster the whole (t, x) plane with the correct slopes (i.e, draw a line of slope $f(t, x)$ at the point (t, x)). Consider

$$x' = x^2 - t.$$

The slopes are zero along the parabola

$$x^2 - t = 0.$$

See Figure 1.

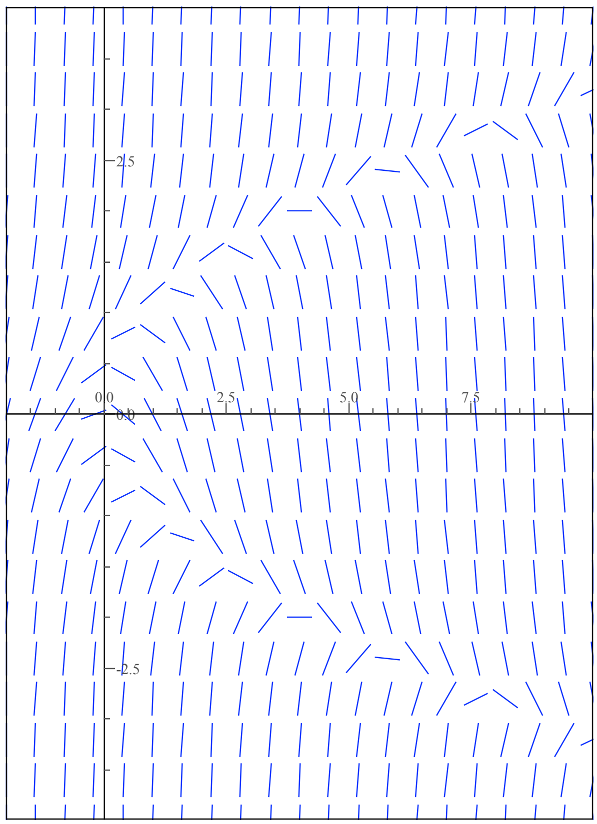

Figure 1: Slope field of $x' = x^2 - t$

Euler's method

What is the computer doing when it draws the slope field? How does it get the numbers?

We step from one one t-value to the next, in steps of size Δt, and estimate the corresponding x-values using the approximation

$$x_{i+1} = x_i + f(t_i, x_i)\Delta t.$$

This makes sense since $f(t_i, x_i)$ is the slope of the graph at the point (t_i, x_i), so this is a linear approximation to the value of x_{i+1}.

Step size	x (using Euler)	error
0.1	11.0462	0.0055
0.05	11.0490	0.0028
0.025	11.0503	0.0014
0.0125	11.0510	0.0007

Table 1: Approximations to $x(1) = 11.0517$ for $x' = 0.1x$, $x(0) = 10$, using Euler's method with different step sizes

Notice that error in Euler's method goes down by about 1/2 every time the step size is halved, for small values of the step size. The error in Euler's method is proportional to the step size in the limit as the step size goes to zero. So, for example, to get an improvement in accuracy of one decimal point, we would need to make 10 times as many calculations.

Heun's algorithm

To do better, we replace the estimate for the derivative in Euler's method with the average of the two values at the beginning and end. However, we have to make sure that this algorithm is not circular in reasoning, so we first get an estimate of the endpoint using Euler's method.

$$x^*_{i+1} = x_i + f(t_i, x_i)\Delta t$$

$$x_{i+1} = x_i + \left(\frac{f(t_i, x_i) + f(t_{i+1}, x^*_{i+1})}{2}\right)\Delta t$$