The Euler Method

The simplest possible integration scheme for the initial-value problem is as the Euler method. Given the differential equation

$$\frac{dy}{dx} = f(x, y),$$

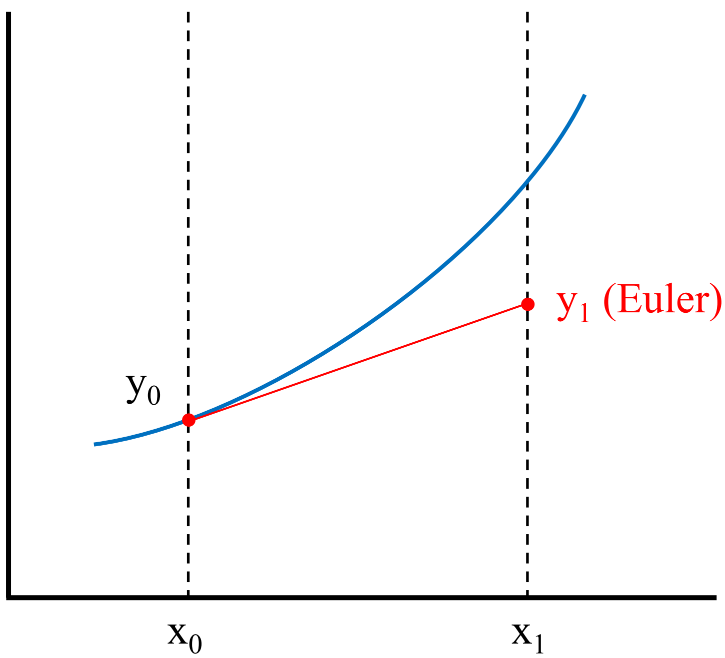
with initial condition $y = y_0$ when $x = 0$, discretize x with

$$x_i = i\delta x, \quad i = 1, 2, \ldots$$

where δx is some suitably short step length. Then, given the numerical value y_i at x_i, we can integrate the system forward according to the prescription

$$\begin{aligned} y_{i+1} &= y_i + \delta x\, f(x_i, y_i) \\ x_{i+1} &= x_i + \delta x. \end{aligned}$$

Geometrically, as illustrated below, we are simply "following the tangent" during the step, then redetermining the new slope of the curve and taking the next step along the new tangent. This scheme generalizes trivially to the case where y is a vector. The programs `euler1.py` and `euler2.py` are simple implementations of the one- and two-dimensional cases.



The Euler method has the undeniable advantage of simplicity. Unfortunately, it is not very accurate—the error at the end of a timestep is $O(\delta x^2)$, and these errors add coherently over the course of the integration. Much worse is the fact that the Euler method can become unstable under certain circumstances—an undesirable property of any integrator. This instability can be controlled by careful stepsize control. However, Euler's low order and the fact that the next simplest method—the Midpoint method—is stable, more accurate, and only marginally more complicated to program, mean that the Euler method is rarely used in real calculations.