Modeling using ODEs: Newton's Law of Cooling and Numerical Methods for solving ODE

Natasha Sharma, Ph.D.

Newton's Law of Cooling

Example

Suppose that in the winter the daytime temperature in a certain office is maintained at 70 degrees F. The heating is shut off at 10 pm and turned on again at 6 am. On a certain day the temperature inside the building at 2 am was found to be 65 degrees F. The outside temperature was 50 degrees at 10 pm and dropped to 40 degrees F by 6 am.
What is the temperature in the building when the heat was turend on at 6 am?

Experimental data: Experiments show that the time rate of change of temperature T of a body B is proportional to the difference between T and the temperature of the surrounding medium.(Newton's Law of Cooling)

$$\frac{dT}{dt} = k(T - T_A)$$

1. How to pick T_A?
   Rule: If we cannot solve the exact mathematical problem, try to solve a simpler problem!
2. What is the form of the ODE?
3. $T(t) = T_A + ce^{kt}$ solves the ODE. (Verify!)

Numeical Solutions to IVP

Suppose we wish to approximate the solutions to the following IVP:

$$\frac{du(t)}{dt} = F(t, u(t)) \tag{1}$$

$$u(0) = u_0, \tag{2}$$

Our task is to obtain a numerical approximation to the solution u to (1)–(2) at some positive time t where $0 \leq t \leq T$.

Numeical Solutions to IVP

Since the computers cannot store or understand continuous time, we need to discretize the time interval $[0, T]$ into say $N+1$ "time screenshots" where we choose to solve for $u(t)$. We set

$$\Delta t = \frac{T}{N}$$

and let $t_0 = 0,\ t_1 = \Delta t,\ t_2 = 2\Delta t,\ cdots t_N = T$. We need to find a way to fill in the table

Time Screenshot	Approximation
$0 = t_0$	$u_0 = u(0)$
$\Delta t = t_1$	$u_1 = u(\Delta t)$
$2\Delta t = t_2$	$u_2 = u(2\Delta t)$
$3\Delta t = t_3$	$u_3 = u(3\Delta t)$
$\vdots$	$\vdots$
$N\Delta t = T$	$u_N = u(T)$

Euler Scheme

$$\frac{du(t)}{dt} = F(t, u(t))$$
$$u(0) = u_0.$$

Replace $\frac{du(t)}{dt}$ by its numerical derivative $D_{\Delta t}(u)(t)$ wrt the discretization of $[0, T]$ that we introduced.

$$D_{\Delta t}(u)(t) \approx \frac{u(t + \Delta t) - u(t)}{\Delta t}$$

so that the Euler Scheme is

$$u(0) = u_0,$$
$$\frac{u(t + \Delta t) - u(t)}{\Delta t} = F(t^*, u(t^*)),$$

where $t^* = t$ or $t^* = t + \Delta t$.

Euler Scheme

$$u(t + \Delta t) = u(t) + \Delta t F(t^*, u(t^*)).$$

If $t^* = t$, the Euler scheme is called Explicit or Forward Euler Scheme.
If $t^* = t + \Delta t$, the Euler scheme is called Implicit or Backward Euler Scheme.
Let us use Maxima to solve the IVPs we have encountered so far.

Euler Scheme: In-Class Activity

1. Download the code `ode_solver.mac`.
2. Your task is to figure out which ODE does this code solve?
3. Does it use Euler Forward or Backward Method?
4. How can you modify the code to solve other ODEs using both the methods for different time steps?