Initial Value Problems & Accuracy

We previously discussed 2 aspects concerning the soundness of a numerical method for solving ODE initial value problems:

→ We first addressed the stability of the solutions of a given ODE $y' = f(t,y)$. For equations with unstable solutions, the methods discussed in our class cannot guarantee that the "computed" solution will stay close to the "real" solution, as $t \to \infty$.

Equations with stable (or, even better, asymptotically stable) solutions have the potential (not a guarantee) for numerical approximations that stay close to the real solution

→ Even when the ODE solutions are stable, a numerical method can still drift away from the real solution if the time step Δt is too large.

e.g. Forward Euler is stable on $y' = \lambda y$ $(\lambda < 0)$ only if $\Delta t < 2/|\lambda|$

Other methods (Backward Euler, Trapezoidal) are unconditionally stable regardless of Δt.

Accuracy is something we examine once we have established that both the ODE has stable solutions, and the numerical method is stable, too.

The analysis of accuracy can address 2 aspects:

→ Global error. Let $y_*(t)$ be the exact solution to $y' = f(t,y)$ and $\hat{y}(t)$ the approximation we construct using a numerical method (e.g. Forward/Backward Euler). If we pick a fixed point in time, say $t_1 \geq t_0$ we ask about the following error value as a function of Δt

$$e = \left| y_*(t_1) - \hat{y}(t_1) \right| = O(\Delta t^d).$$

Ideally, we want our method to offer an increasingly better approximation of $\hat{y}(t_1)$ as $\Delta t \to 0$

If d is the exponent that satisfies $e = O(\Delta t^d)$, the method is said to be d-order accurate.

We will see that $\left.\begin{array}{l}\text{Forward Euler}\\ \text{Backward Euler}\end{array}\right\} \Rightarrow$ are 1st order accurate

Trap. Rule is 2nd order accurate.

→ Local error: In this variant, we make the assumption that the k-th approximation y_k produced by our method happens to be exact (e.g $y_k \equiv y_{exact}(t_k)$).

(of course, this is not something we realistically expect to happen, but we take it as a best-case-scenario anyways).

Then, we examine the discrepancy between the next value produced by our method. (y_{k+1}) and the exact value. $y_{exact}(t_{k+1})$

$$e_k = \left| y_{k+1} - y_{exact}(t_{k+1}) \right| = O(\Delta t^{d+1})$$

Theorem When the local error is $e_k = O(\Delta t^{d+1})$, the global error is $e = O(\Delta t^d)$ (one power less) and the method is thus d-order accurate.

Although the global error is a bit challenging to compute, the local error is more easily approximated. On the model equation $y' = \lambda y$ ($\lambda < 0$) we have the exact solution as $y_{exact}(t) = ce^{\lambda t}$. Thus:

$$y_{exact}(t_{k+1}) = ce^{\lambda(t_{k+1})} = ce^{\lambda(t_k + \Delta t)} = ce^{\lambda t_k} \cdot e^{\lambda \Delta t} = y_k \cdot e^{\lambda \Delta t}$$

since we assumed that y_k is exact!!

However, for the y_{k+1} produced by a numerical method, we have

$$y_{k+1} = \begin{cases} (1+\lambda\Delta t)\, y_k & \text{for Forward Euler} \\ \dfrac{1}{1-\lambda\Delta t}\, y_k & \text{for Backward Euler} \\ \dfrac{1+\lambda\Delta t/2}{1-\lambda\Delta t/2}\, y_k & \text{for Trapezoidal Rule.} \end{cases}$$

At this point, it is useful to remember the following infinite series:

$$e^x = 1 + x + \frac{x^2}{2} + \frac{x^3}{3!} + \cdots + \frac{x^k}{k!} + \cdots \qquad (1)$$

$$\frac{1}{1-x} = 1 + x + x^2 + x^3 + x^4 + \cdots \qquad (2)$$

$$\frac{1+x}{1-x} = 1 + 2x + 2x^2 + 2x^3 + 2x^4 + \cdots \qquad (3)$$

Start with Forward Euler. We have:

$$e_k = |y_{k+1} - y_{exact}(t_{k+1})| = |(1+\lambda\Delta t) y_k - e^{\lambda\Delta t} y_k| =$$

$$= |1+\lambda\Delta t - e^{\lambda\Delta t}|\,|y_k|.$$

Using (1), with $x = \lambda\Delta t$ we get:

$$e^{\lambda\Delta t} = 1 + \lambda\Delta t + \frac{(\lambda\Delta t)^2}{2} + \frac{(\lambda\Delta t^3)}{3!} + \cdots$$

$$\Rightarrow \quad e^{\lambda\Delta t} - (1+\lambda\Delta t) = \frac{(\lambda\Delta t)^2}{2!} + \underbrace{\frac{(\lambda\Delta t)^3}{3!} + \frac{(\lambda\Delta t)^4}{4!} + \cdots}$$

All these terms are insignificant compared to Δt^2 when $\Delta t \to 0$.

Thus $\left| (1+\lambda\Delta t) - e^{\lambda\Delta t} \right| = \frac{\lambda^2}{2}\Delta t^2 + (\text{small terms}) = O(\Delta t^2)$

Thus $e_k = O(\Delta t^2) \cdot |y_k| = O(\Delta t^2)$

$\Rightarrow$ Local error is $O(\Delta t^2)$
Global error is $O(\Delta t)$
Forward Euler is 1st order accurate!

For Backward Euler, we have

$$e_k = \left| \frac{1}{1-\lambda\Delta t} - e^{\lambda\Delta t} \right| |y_k|$$

using (1) & (2) with $x = \lambda\Delta t$:

$$\frac{1}{1-\lambda\Delta t} = \cancel{1} + \cancel{(\lambda\Delta t)} + (\lambda\Delta t)^2 + (\lambda\Delta t)^3 + \cdots$$

$$e^{\lambda\Delta t} = \cancel{1} + \cancel{(\lambda\Delta t)} + \frac{(\lambda\Delta t)^2}{2} + \frac{(\lambda\Delta t)^3}{3!} + \cdots$$

$$\frac{1}{1-\lambda\Delta t} - e^{\lambda\Delta t} = \frac{\lambda^2}{2}\Delta t^2 + (\text{small terms}) = O(\Delta t^2)$$

$\Rightarrow e_k = O(\Delta t^2)$ and Backward Euler is still <u>first order</u>.

However, for trapezoidal rule, we get:

$$e_k = \left| \frac{1 + \lambda\Delta t/2}{1 - \lambda\Delta t/2} - e^{\lambda\Delta t} \right| |y_k|$$

$$\frac{1+\frac{\lambda\Delta t}{2}}{1-\frac{\lambda\Delta t}{2}} = \cancel{1} + \cancel{2\left(\frac{\lambda\Delta t}{2}\right)} + \cancel{2\left(\frac{\lambda\Delta t}{2}\right)^2} + 2\left(\frac{\lambda\Delta t}{2}\right)^3 + \cdots$$

$$e^{\lambda\Delta t} = \cancel{1} + \cancel{\lambda\Delta t} + \cancel{\frac{(\lambda\Delta t)^2}{2}} + \frac{(\lambda\Delta t)^3}{6} + \cdots$$

$$\Rightarrow \frac{1+\lambda\Delta t/2}{1-\lambda\Delta t/2} - e^{\lambda\Delta t} = \frac{(\lambda\Delta t)^3}{4} - \frac{(\lambda\Delta t)^3}{6} + (\text{small terms}) = O(\Delta t^3)$$

Thus $e_k = O(\Delta t^3)$ and consequently trapezoidal rule is

<u>2nd order accurate!</u>

Although we only used the model equation $y' = \lambda y$ to examine accuracy, our findings for these methods extend to general equations $y' = f(t, y)$ (as long as they have stable solutions!).

Introduction to systems of equations

We defined an initial value problem as:

$$y' = f(t, y)$$
$$y(t_0) = y_0$$

Here, we only had one unknown function and one differential equation. It is possible however to define an entire system of coupled system of ODEs as follows:

n unknown functions : $y_1(t), y_2(t), \dots, y_n(t)$

n ODEs:

$$y_1'(t) = f_1(t, y_1, y_2, \dots, y_n)$$
$$y_2'(t) = f_2(t, y_1, y_2, \dots, y_n)$$
$$\vdots$$
$$y_n'(t) = f_n(t, y_1, y_2, \dots, y_n)$$

and, finally, n initial conditions

$$y_1(t_0) = c_1$$
$$y_2(t_0) = c_2$$
$$\vdots$$
$$y_n(t_0) = c_n$$