Lecture 7 : Numerical Methods

So far, the methods for solving 1^{st}-order ODEs have been:

- Separable ODEs

$$\frac{dy}{dx} = g(x)p(y)$$

$$\Rightarrow \int \frac{1}{p(y)}\,dy = \int g(x)\,dx$$

- Linear ODEs

$$\frac{dy}{dx} + p(x)y = g(x) \quad \Rightarrow \quad \mu(x) = e^{\int p(x)\,dx}$$

$$y(x) = \frac{1}{\mu(x)} \int \mu(x) g(x)\,dx$$

Ex $\quad y' + x^2 y = 0$

Separable: $\rightarrow \quad \frac{dy}{dx} = -x^2 y$

$$\int \frac{1}{y}\, dy = -\int x^2 dx$$

$$\Rightarrow \quad \ln(|y|) = -\frac{1}{3}x^3 + C$$

Implicit Form $\quad \ln(|y|) + \frac{1}{3}x^3 = C$

Explicit Form $\quad y = e^{-\frac{1}{3}x^3 + C} = A e^{-\frac{1}{3}x^3}$

Or... Linear

$$p(x) = x^2 \Rightarrow \mu = e^{\int x^2 dx} = e^{\frac{1}{3}x^3}$$

$$\rightarrow \quad \frac{d}{dx}(\mu y) = \mu g = 0$$

$$\mu y = C$$

$$\Rightarrow \quad e^{\frac{1}{3}x^3} y = C$$

$$\Rightarrow \quad y = C e^{-\frac{1}{3}x^3}$$ ✓ Same

$$\left(\begin{array}{c} \text{Integrating Zero} \\ \int 0 \, dx = C \\ \int_a^b 0 \, dx = 0 \end{array} \right)$$

Are there other methods?

Yes... but not many

- Substitutions
- Exact ODEs

What are numerical methods

So far, most techniques you've learned have been "analytic"

$\Rightarrow$ symbolic manipulation

Ex.

$$ax^2 + bx + c = 0 \rightarrow x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

How would we approach ODEs numerically?

$$\frac{dy}{dx} = \lim_{h \to 0} \frac{y(x+h) - y(x)}{h}, \quad h = \Delta x$$

$$\approx \frac{y(x+h) - y(x)}{h},$$

$$0 < h \ll 1$$

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

$$\frac{y(x+h) - y(x)}{h} \approx f(x, y(x))$$

$$\Rightarrow \quad y(x+h) - y(x) \approx h f(x, y(x))$$

$$\Rightarrow \quad y(x+h) \approx y(x) + h f(x, y(x))$$

"Forward Euler"
(pronounced "oiler")

<u>Example 1</u> $\quad \frac{dy}{dx} = -y \, , \quad y(0) = 1$

(answer: $y(x) = e^{-x}$)

what is $y(0.1)$?

Set $x=0$, $h = 0.1$

$$y(x+h) = y(x) + h\, f(x, y(x))$$

Here $f(x,y) = -y$

$$y(0+0.1) \approx y(0) + 0.1\,(\underbrace{-y(0)}_{f})$$

$$\Rightarrow y(0.1) \approx 1 + 0.1\,(-1) = 0.9$$

Actual answer $y(0.1) = e^{-0.1} \approx 0.9048\ldots$

How do we cut down on error?

Reduce $h = \Delta x$!

Instead pick $x=0$, $h = 0.05$

$$\begin{aligned} y(x+h) &= y(x) + h\, f(x,y) \\ &= y(x) + h\,(-y(x)) \\ &= y(x) - h\, y(x) \\ &= (1-h)\, y(x) \end{aligned}$$

$$y(0+0.05) \approx (1-0.05)\, y(0)$$

$$\Rightarrow \quad y(0.05) \approx (0.95)(1) = 0.95$$

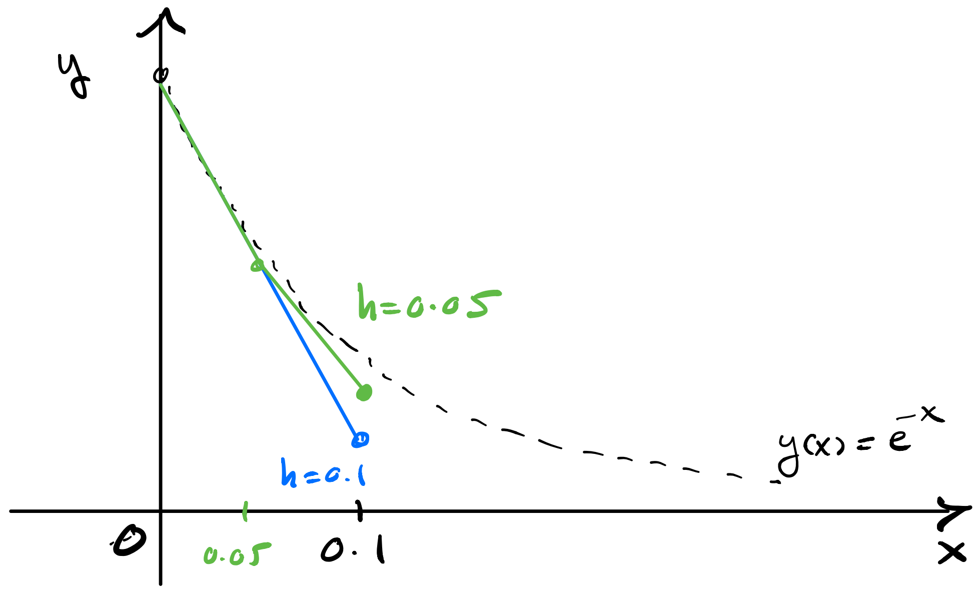
Now, set $x = 0.05$, $h = 0.05$

$$y(0.05+0.05) \approx (1-0.05)\, y(0.05)$$

$$\Rightarrow \quad y(0.1) \approx (0.95)(0.95) = 0.9025$$

(answer $y(0.1) = e^{-0.1} \approx 0.9048$)

cut h in half $\Rightarrow$ error cut in half



Why not set $h = 0.01$

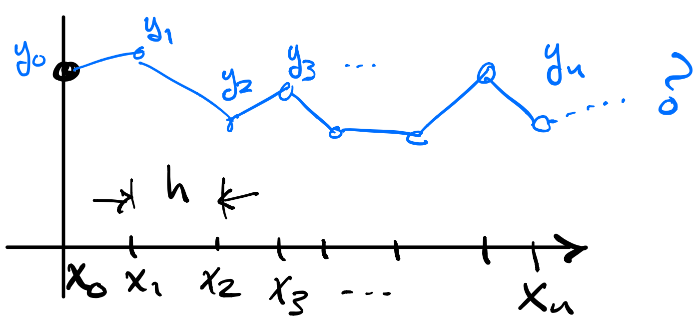
10 times
$$\left\{\begin{array}{l} y(0.01) \approx (1-0.01)\, y(0) = 0.99 \\ y(0.02) \approx (1-0.01)\, y(0.01) = 0.9801 \\ \quad \vdots \\ y(0.1) \approx (1-0.01)\, y(0.09) \approx 0.90438 \end{array}\right.$$

Exact answer $e^{-0.1} \approx 0.9048\ldots$

In practice : Computers Do This

Write this as an algorithm!

$$\frac{dy}{dx} = f(x,y) \quad , \quad y(x_0) = y_0$$



$$x_{n+1} = x_n + h \quad , \quad h = \Delta x$$

$$y(x_n) = y_n$$

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

Forward Euler Method

Disadvantages

1) Need a computer & know how to code (C++, Fortran, Python, MatLab, ...)

2) Calculations are specific to an initial condition

3) Always <u>some</u> error

Cutting down on error:

A) Smaller step size ($h = \Delta x$)

B) <u>Smarter</u> algorithm