Ordinary Differential Equations

EAI Math Reading Group

18/02/2024

Math anniversaries of the day

Sophus Lie (died 1899)

Carl Gustav Jacob Jacobi (died 1851)

Image source: Wikimedia

Outline

Today: ODE 101 any%

1. What is an ODE?
2. Linear ODEs and linearization
3. Neural ODEs

Differential Equation

A *functional* equation involving derivatives

$$\frac{dx}{dt}(t) = 2\cos(t) - x(t).$$

A *solution* of the equation is a differentiable function $x(t)$ that satisfies the equation, e.g.

- $x(t) = \cos(t) + \sin(t)$
- $x(t) = \cos(t) + \sin(t) + e^{-t}$
- $x(t) = \cos(t) + \sin(t) + Ce^{-t}$

When multiple solutions exists, we can nail it down to one using *Boundary conditions*, e.g. $x(0) = 0$.

Differential Equation Zoo

Ordinary Differential Equations (ODE):

- $\frac{dx}{dt} = f(x,t)$ (first order ODE)
- $\frac{dx}{dt} = kx$ (exponential growth/ decay)
- $\frac{dx}{dt} = k(A - x)$ (Newton's law of cooling)
- $m\frac{d^2x}{dt^2} + c\frac{dx}{dt} + kx = f(t)$ (dampened oscillator with driving force)

NB. common notations: $\dot{x} = \frac{dx}{dt}$, $y' = \frac{dy}{dx}$

Partial Differential Equations (PDEs)

- $\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0$ (transport equation)
- $\frac{\partial u}{\partial t} + \frac{\partial^2 u}{\partial x^2} = 0$ (Heat equation == Diffusion)
- $\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$ (wave equation)

The blowup problem[1]

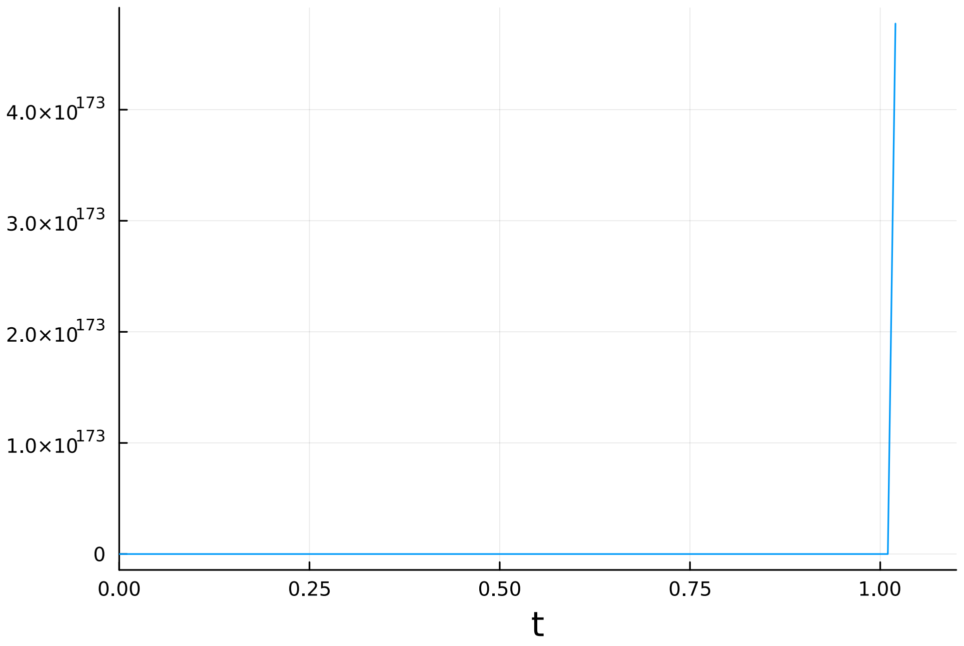
Consider

$$\dot{x} = x^2$$

General solution:

$$x(t) = \frac{1}{C - t}$$

The solution goes to infinity in finite time!



[1]literally the singularity

Fundamental Equations

$$\dot{y} = \alpha y$$

General solution:

$$y(t) = Ce^{\alpha t}$$

- $\alpha > 0$: exponential growth
- $\alpha < 0$: exponential decay

$$\ddot{y} = -k^2 y$$

General solution:

$$y(t) = C_1 \cos(kt) + C_2 \sin(kt)$$

Solving ODEs manually

TLDR: Don't

- Standard ODE courses typically teach various methods handed down for generations
- Almost all interesting (nonlinear) ODEs usually can't be solved analytically
- Knowing the basic (e.g. linear) ODEs is good enough for qualitative analysis

Solving ODEs numerically: Euler's method

A simple scheme for numerically integrating an ODE uses the finite difference formula for the derivative : $f'(t) \approx \frac{f(t+h)-f(t)}{h}$

$$\Rightarrow f(t+h) \approx f(t) + hf'(t)$$

Taking a small enough *step size* $h > 0$, this yields Euler's method

$$y_{k+1} = y_k + hf(y_k, t_k),$$

for $t_k = kh$.

Euler's method is a *first order method* (e.g. reducing h by reduces the *error* by half)

This is *suboptimal*. Usually better to use higher order methods (e.g. RK4)

Intermezzo: Gradient flow

Consider the first order ODE

$$\dot{x} = -\nabla f(x),$$

with $x(t) \in \mathbb{R}^n$ and $f \in C^1(\mathbb{R}^n, \mathbb{R})$

Applying Euler's method to this yields

$$x_{k+1} = x_k - h\nabla f(x_k)$$

It's just Gradient Descent!

Linear ODEs

Consider ODEs of the form

$$\dot{x} = Ax$$

with $x(t) \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n \times n}$

Based on the 1D case, we'd expect something like

$$x(t) = e^{At} x(0)$$

But does "e^{At}" make sense as an expression?

Higher order linear ODEs

Consider the n-th order ODE

$$a_n \frac{d^n x}{dt^n} + a_{n-1} \frac{d^{n-1} x}{dt^{n-1}} + ... + a_1 \frac{dx}{dt} + a_0 x + b = 0$$

where $a_n, ..., a_0, b$ are some (mostly) arbitrary functions.

We can turn this into a n-dimensional *first order* ODE by

$$\frac{dy_k}{dt} = y_{k+1} \ (0 \leq k < n)$$

$$\frac{dy_n}{dt} = \sum_{k=0}^{n-1} \frac{a_k}{a_n} y_k + \frac{b}{a_n}$$

Matrix exponential

We can define what the *matrix exponential* means by using Taylor series

$$e^A = \sum_{k=0}^{\infty} \frac{A^k}{k!}$$

This is well-defined for any matrix $A \in \mathbb{R}^{n \times n}$, and we can do even better. If A can be diagonalized as $A = V \Lambda V^{-1}$:

$$e^A = V e^{\Lambda} V^{-1}$$

where

$$e^{\Lambda} = \begin{pmatrix} e^{\lambda_1} & & \\ & \ddots & \\ & & e^{\lambda_n} \end{pmatrix}$$

Matrix exponential (continued)

For a given A, the eigenvalues tell us about the behaviour of the trajectories:

- $\lambda > 0$: unstable mode $\rightarrow$ goes to infinity
- $\lambda < 0$: stable mode $\rightarrow$ decays to zero
- $\lambda = 0$: stays constant in that direction
- $\lambda \in \mathbb{C}$: oscillatory behaviour

Neural ODEs

People started using residual connections like

$$h_{t+1} = h_t + f(h_t, \theta_t)$$

If you squint a little bit, this is just Euler's method.

$\rightarrow$ layer of the form

$$\dot{h}(t) = f(h(t), \theta, t)$$

where f is a neural network with parameters θ.

Neural ODEs

Some advantages

- Adjoint method $\Rightarrow$ efficient backprop, can treat the solver as a black box
- Natively handles irregularly sampled data
- Adaptive solvers $\Rightarrow$ "infinite depth"/adaptive depth
- ...

Lots of applications: Diffusion models, Scientific ML/PINNs (inject NNs inside traditional physics models)

Where do we go from here?

Plenty of options

- State-Space Models
- Reservoir Computing/Liquid State Networks
- Basic Chaos Theory
- …