Lecture 5 - 2/12

- Intro to Hamiltonian dynamics
- Symplectic structure
- Machine learning (continuous) Hamiltonians
- the discrete gradient method

In Mondays lecture, we stepped through coding up a generic FD nonlinear stencil fitter

It worked (most of the time) - but why?

Today we'll talk about building in energy conservation exactly

<u>Lemma</u> If $A = -A^T$, $x^T A x = 0 \quad \forall x$

<u>Pf</u> $x^T A x = \frac{1}{2} x^T (A + A) x = \frac{1}{2} x^T (A - A^T) x$

$= \frac{1}{2} x^T (A - A) x = 0$

Conservation of Energy

Let $q \in \mathbb{R}^N$ be vector of generalized position

$p \in \mathbb{R}^N$ be vector of generalized momentum

$x = \begin{pmatrix} q \\ p \end{pmatrix} \in \mathbb{R}^{2N}$ be the state of system

The system is canonically Hamiltonian if

$$\frac{dp}{dt} = -\frac{\partial H}{\partial q} \qquad \text{for } H(q,p) \in \mathbb{R}$$

$$\frac{dq}{dt} = \frac{\partial H}{\partial p}$$

Thm A canonically Hamiltonian system conserves H, i.e

$$\frac{dH}{dt} = 0$$

Pf

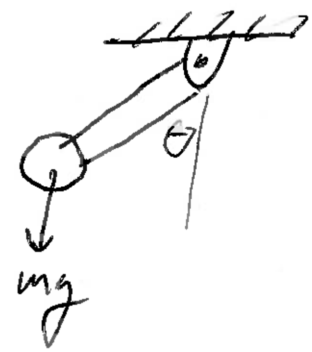
$$\frac{dH}{dt} = \frac{\partial H}{\partial p}\frac{dp}{dt} + \frac{\partial H}{\partial q}\frac{dq}{dt}$$

$$= -\frac{\partial H}{\partial p}\frac{\partial H}{\partial q} + \frac{\partial H}{\partial q}\frac{\partial H}{\partial p}$$

$$= 0$$

∎

Ex Nonlinear pendulum



- Newton

$$m\ddot{x} = -mg \sin\theta$$

$$\text{s.t. } \|x\| = L$$

- Can enforce constraint by switching to polar coordinates and assuming $x = L\,\theta(t)$

$$mL\ddot{\theta} = -mg\sin\theta$$

$$\ddot{\theta} = -\lambda^2 \sin\theta \quad , \quad \lambda = \sqrt{\frac{g}{L}}$$

Small θ limit

$$\sin\theta \approx \theta$$

$$\ddot{\theta} = -\lambda\theta$$

Letting

$$q = \theta$$

$$p = \dot{\theta}$$

$$H = \frac{1}{2}p^2 + \frac{\lambda^2}{2}q^2$$

Then

$$\partial_p H = p \quad , \quad \partial_q H = \lambda^2 q$$

$$\frac{dp}{dt} = -\partial_q H = -\lambda^2 q$$

$$\frac{dq}{dt} = \partial_p H = q$$

For the nonlinear case, there is a trick to identify Hamiltonians of the form of dynamics

$$\ddot{\theta} + F(\theta) = 0$$

if F is a function w/ simple antiderivative

$$\frac{d}{dt}\left[\frac{1}{2}\dot{\theta}^2 + \int^{\theta(t)} F(\phi)\, d\phi\right] = \left[\ddot{\theta} + F(\theta)\right]\dot{\theta}$$

Recall

$$\frac{d}{dx}\int_a^x f = f(x)$$

Motivated by this, multiply our eqn by $\dot{\theta}$

$$\ddot{\theta}\dot{\theta} + \lambda^2 \dot{\theta} \sin\theta = 0$$

$$= \frac{d}{dt}\left(\frac{1}{2}\dot{\theta}^2 - \lambda^2 \cos\theta\right) = 0$$

$$\Rightarrow \quad \frac{1}{2}\dot{\theta}^2 - \lambda^2 \cos\theta = \text{const.}$$

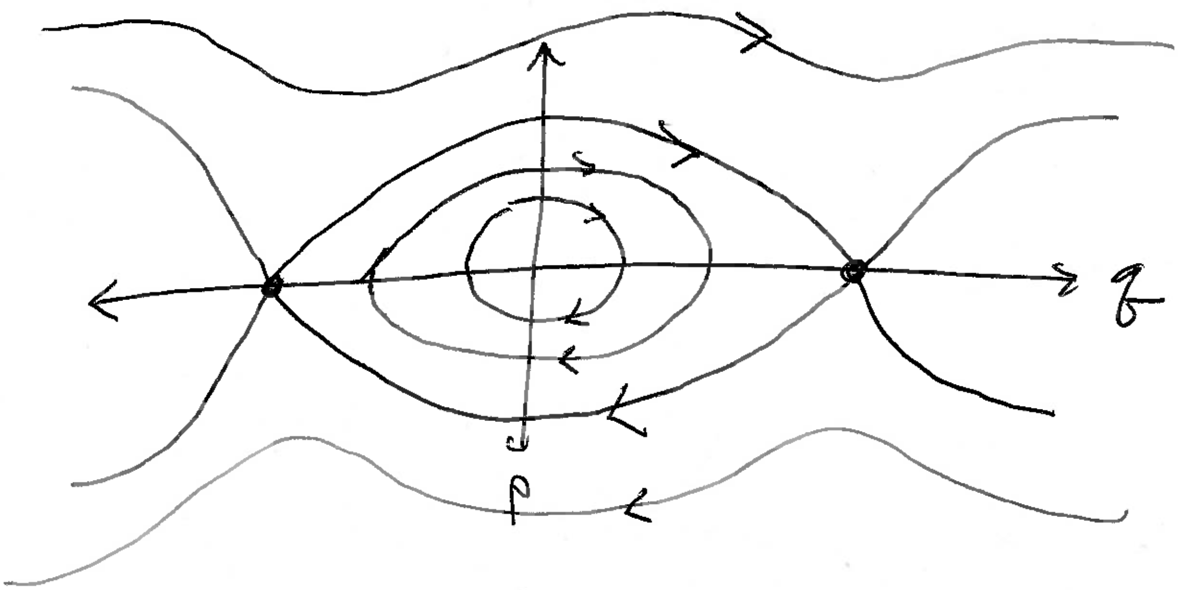
Taking $H = \frac{1}{2}p^2 - \lambda^2 \cos q$

$$\frac{dp}{dt} = -\partial_q H = -\lambda^2 \sin q$$

$$\frac{dq}{dt} = \partial_p H = p$$

- H is often called an integral or invariant of the dynamics
- The whole game is to reverse engineer an H which, for a good choice of p, q, gives our dynamics (typically an energy)
- This is hard in general, and will become involved for PDEs
- This is <u>not</u> hard if we machine learn H, and use it to steer the system dynamics

Phase Diagram of Soln



Qualitative features of soln

- Separatrix denotes switch between small rocking oscillations and pendulum flying around forever
- Inner loops are <u>closed</u> so that solutions should repeat <u>forever</u>

We'd like to be able to recover this in an ML model

Symplectic structure

Note that we can compactly write

$$\dot{X} = S(x) \nabla_x H$$

$$S(x) = \begin{pmatrix} 0 & -I \\ I & 0 \end{pmatrix}, \quad \nabla_x H = \begin{pmatrix} \partial_p H \\ \partial_q H \end{pmatrix}$$

Recall the Gauss div theorem

$$\int_\Omega \nabla \cdot F = \int_{\partial\Omega} F \cdot dA$$

Taking $F = x$, $x \in \mathbb{R}^d$

$$\int_{\partial\Omega} x \cdot dA = \int_\Omega \nabla \cdot x = d|\Omega|$$

So that $\int x \cdot dA$ corresponds to the area of Ω times dimension

$$\frac{d}{dt} \int_{\partial\Omega} x \cdot dA = \int_{\partial\Omega} \dot{x} \cdot dA$$

$$= \int_{\partial\Omega} S \nabla H \cdot dA$$

$$= \int_\Omega \nabla \cdot S \nabla H \, dx$$

$$= \sum_{i,j} \int \partial_{q_i} \partial_{p_j} H - \partial_{p_i} \partial_{q_j} H \, dx$$

$$= 0$$

Thus, area in phase space is conserved

This exactly the property that's violated when we over damp our system

Non-Canonical Hamiltonian

In general, we can consider arbitrary S.

$$\dot{x} = S(x) \nabla_x H$$

$$S(x) = -S(x)$$

S is often referred to as a Poisson matrix

Thm A non-canonical Hamiltonian preserves H

Pf
$$\begin{aligned}
\frac{dH}{dt} &= \nabla_x H^T \dot{x} \\
&= \nabla_x H^T S(x) \nabla_x H \\
&= \frac{1}{2} \nabla_x H^T S(x) \nabla_x H + \frac{1}{2} \nabla_x H^T S(x)^T \nabla_x H \\
&= \frac{1}{2} \nabla_x H^T S(x) \nabla_x H - \frac{1}{2} \nabla_x H^T S(x) \nabla_x H \\
&= 0
\end{aligned}$$

Machine learning dynamics

In continuum setting we can easily learn a Hamiltonian

$$Q = NN_1(x), \quad NN_1 : \mathbb{R}^N \to \mathbb{R}^{N \times N}$$

$$S(x) = Q - Q^T$$

$$H = NN_2(x)$$

$$\Rightarrow \quad \dot{x} = \left(NN_1(x) - NN_1(x)^T\right) \nabla_x NN_2(x)$$

But to fit to data, we need to finite difference $\dot{x} = \frac{x^{n+1} - x^n}{k}$, which will either <u>grow</u> area in phase space (EE) or <u>shrink</u> (IE)

To discretely conserve H, we turn to "geometric integration" theory

<u>Ref</u> "Geometric Numerical Integration"
Hairer, Lubich, Wanner

Discrete Gradient Method

"On the construction of discrete gradients" Mansfield, Quispel

Consider $\dot{x} = S(x)\,\nabla H(x)$ (I'll drop x-dep)

Discretize $$\frac{x^{n+1} - x^{n}}{K} = \tilde{S}(x^{n+1}, x^{n})\ \overline{\nabla} H(x^{n+1}, x^{n})$$

skew-symm

where
$$\lim_{x^{n+1}\to x^{n}} \tilde{S}(x^{n+1}, x^{n}) = S(x^{n})$$
$$\lim_{x^{n+1}\to x^{n}} \overline{\nabla} H(x^{n+1}, x^{n}) = \nabla H(x^{n})$$
consistency

and $$(x^{n+1} - x^{n}) \cdot \overline{\nabla} H = H(x^{n+1}) - H(x^{n})$$
$$\overline{\nabla} H(x^{n}, x^{n}) = \nabla H(x^{n})$$

Then $$H(x^{n+1}) - H(x^{n}) = 0$$

pf $$H(x^{n+1}) - H(x^{n}) = \overline{\nabla} H^{T} (x^{n+1} - x^{n})$$
$$= \overline{\nabla} H^{T} \frac{\tilde{S}(x^{n+1}, x)}{K} \overline{\nabla} H$$
$$= 0$$

This gives a "wish list" for how to build $\tilde{S}$, $\overline{\nabla} H$

Ex 1 Harten, Lax, Van Leer, 1983

Denote

$$\nabla H = (H_{x_1}, \dots H_{x_n}), \quad H_{x_i} = \frac{\partial}{\partial x_i} H$$

Define

$$\overline{\nabla} H(x,y) = (\overline{H}_{x_1}, \dots \overline{H}_{x_n})$$

$$\overline{H}_{x_i} = \int_0^1 H_{x_i}\big((1-\xi)x + \xi y\big)\, d\xi$$

Then

$$\overline{\nabla} H(x,y) \cdot (y-x)$$

$$= \sum_{i=1}^{n} \overline{\nabla} H_i(x,y)(y_i - x_i) = \sum_{i=1}^{n} \int_0^1 H_{x_i}\big[(1-\xi)x + \xi y\big](y_i - x_i)\, d\xi$$

Reverse the chain rule

$$= \int_0^1 \frac{d}{d\xi} H\big[(1-\xi)x + \xi y\big]\, d\xi$$

$$= H(y) - H(x)$$

Ex 2 Itoh & Abe, 1988

Ex 3 Gonzalez, 1996