University of BRISTOL

[2020-09-07 Mon]

Proposal on machine learning via DS

What and why?

- ML is ‘fancy’ model fitting
- ODE solutions could be a rich source of models to fit

Section 1: formulation

- Let $z(T, x)$ be a solution to an ODE, evaluated at time T, initial condition x

Section 1: formulation

- Let $z(T,x)$ be a solution to an ODE, evaluated at time T, initial condition x
- $\frac{\mathrm{d}z}{\mathrm{d}t} = f(A(t),z), \;\; z(0)=x$

Section 1: formulation

- Let $z(T, x)$ be a solution to an ODE, evaluated at time T, initial condition x
- $\frac{\mathrm{d}z}{\mathrm{d}t} = f(A(t), z), \;\; z(0) = x$
- $u(x) = \mathbf{a} \cdot \mathbf{z} + b$

Section 1: formulation

- Let $z(T, x)$ be a solution to an ODE, evaluated at time T, initial condition x
- $\frac{\mathrm{d}z}{\mathrm{d}t} = f(A(t), z), \;\; z(0) = x$
- $u(x) = \mathbf{a} \cdot \mathbf{z} + b$
- $\mathrm{argmin}_{\mathbf{a},b,A} \sum (y_i - u(x_i))^2$

Finding a controller

- $\text{performance} = \int \text{error}^2 \mathrm{d}\mu(x)$
- $\frac{\mathrm{d}\ \text{performance}}{\mathrm{d}A} = \int \frac{\mathrm{d}\ \text{error}^2}{\mathrm{d}A} \mathrm{d}\mu(x)$
- $\frac{\mathrm{d}\ \text{perturbation}}{\mathrm{d}\tau} = J_z \text{perturbation}$
- $\frac{\mathrm{d}\ \text{output}}{\mathrm{d}A} = \text{perturbation}(T)$

Connection to deep NNs

- Deep NNs are a dynamical system that can change dimension
- Continuous NNs cannot change dimensions
- Continuous NNs can overcome issues with training deep NNs

Connection to deep resnets

- Residual neural networks overcome vanishing gradients by selectively omitting layers
- The dynamical systems viewpoint explains why this should help training
- Resnets learn an Euler-discretisation of an ODE

Representability and controllability

- We need to be sure that the flow map can process data as desired
- This is a problem of controllability
- Idealised problem: can the flow-map model arbitrary mappings on the data?

Continuum in space

- PDE models are useful when we have spatially structured data

- Using a convolutional kernel gives CNN-like behaviours

Constraints, structure, and regularisation

- We could add constraints to the system
- We could add structure to the ODEs
- We could add regularisation terms

Clustering and density estimation

- A clustering model is presented
    - Not quite sure how it relates to the rest of the paper?

- Density estimation can also be performed with the flow-map framework

Paper suggestion

Raissi, Maziar, Paris Perdikaris, and George E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations." Journal of Computational Physics 378 (2019): 686-707.

Any volunteers?