Operator Learning: An Overview

Fall 2022 Directed Reading Program
University of Pennsylvania
Mentee: Joshua Anumolu
Mentor: Leonardo Ferreira Guilhoto

Definition

01

Motivation

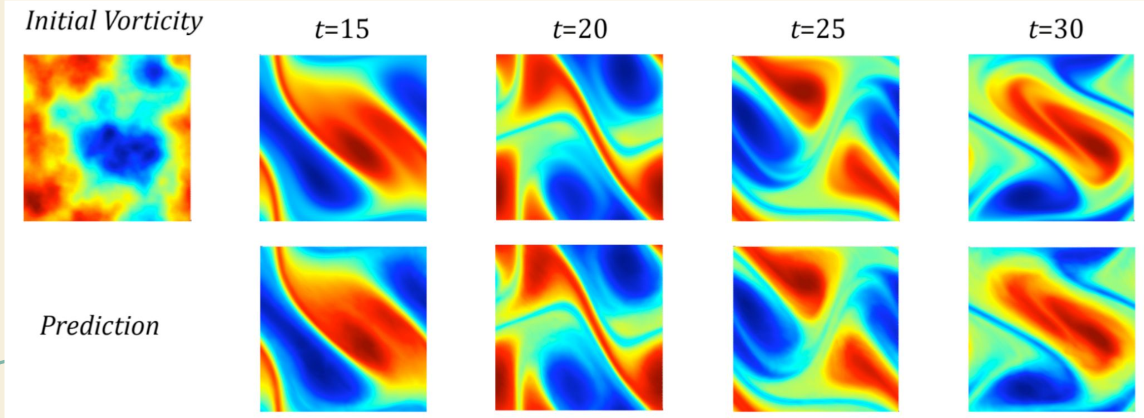
Operator maps between function spaces: *G: X→ Y*
Compute *G(u)(y)* for any *u* ϵ X and any *y* in domain of G(u).

Example:

f(x) = sin(x), y = 0
G(f)(y) = f'(y) = cos(0) = 1

The heat equation

$$\frac{\partial u}{\partial t} = \alpha\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) = \alpha\nabla^2 u$$



Difficulties

Function spaces are infinite dimension; how do encode functions?

Structure

02

Abstract Architecture

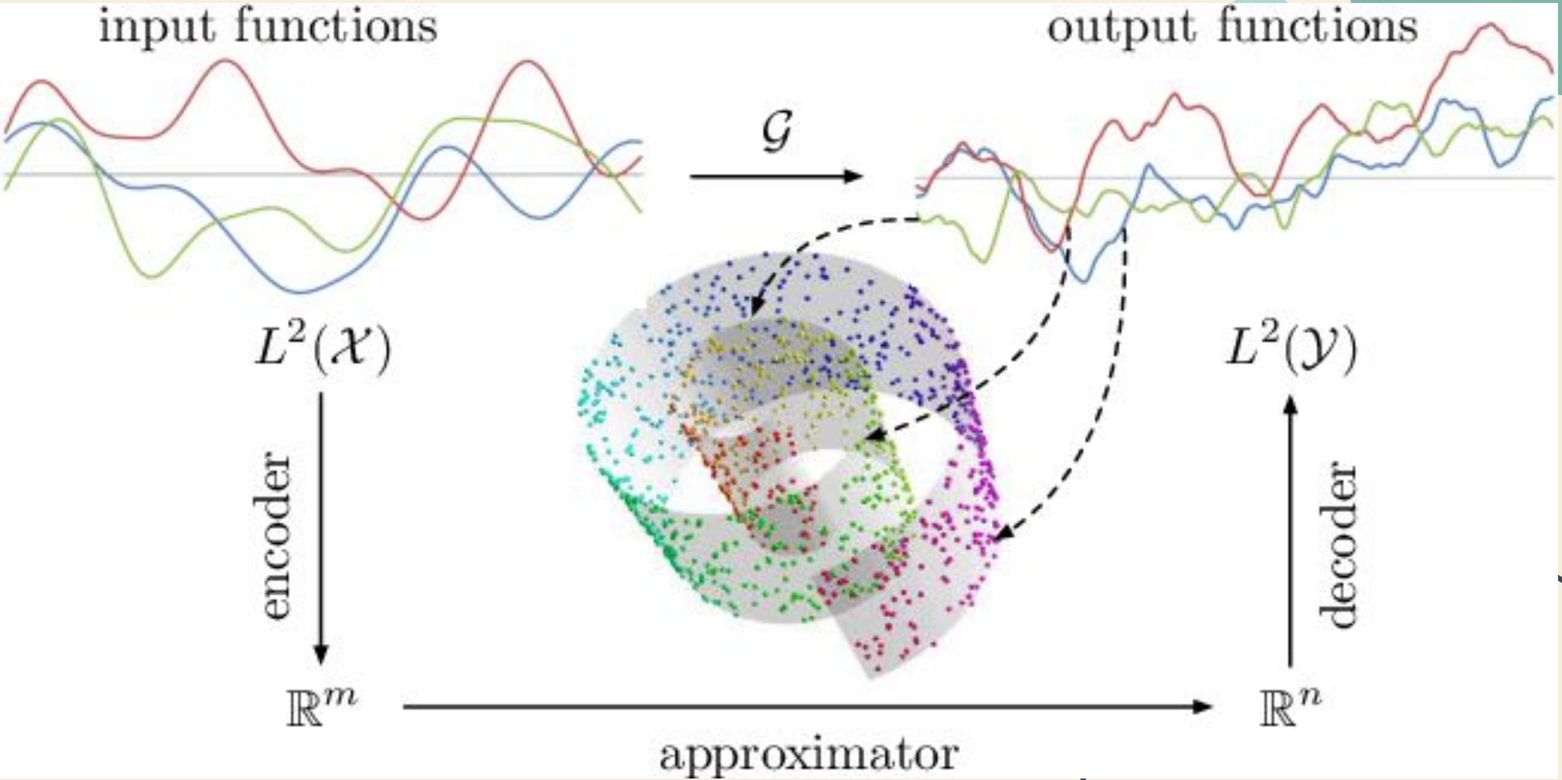


Image from Seidman et. al. (2022).

3 Maps: Encoder, Approximator, Decoder

$$\mathcal{G} \approx \mathcal{F} := \mathcal{D} \circ \mathcal{A} \circ \mathcal{E}.$$

$$\begin{tikzcd}
L^2(\mathcal{X}) \arrow[r, "\mathcal{G}"] \arrow[d, "\mathcal{E}"] & L^2(\mathcal{Y}) \\
\mathbb{R}^m \arrow[r, "\mathcal{A}"] & \mathbb{R}^n \arrow[u, "\mathcal{D}"']
\end{tikzcd}$$

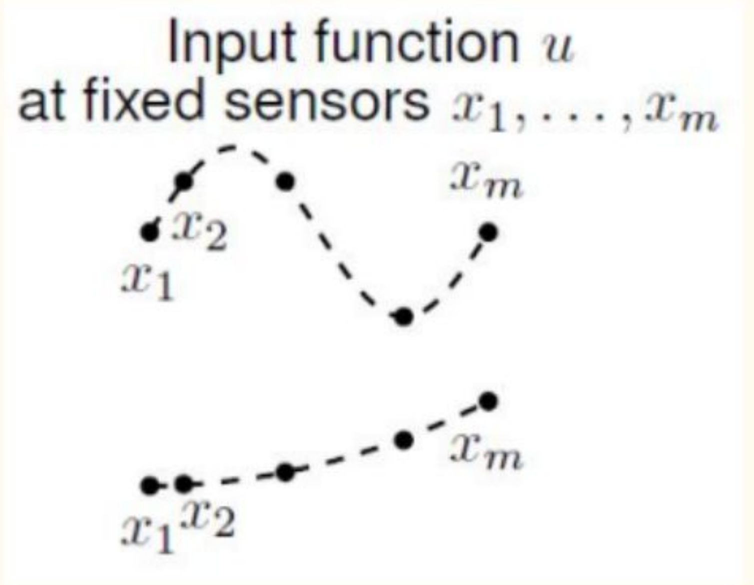
Image from Seidman et. al. (2022).

Encoder

u **is an infinite dimensional function. How do we encode u?**

- Store its values at series of points (sensors)
- Store as vector

Ie, f(0), f(π/2) f(π), = sin(0), sin(π/2), sin(π) = 0, 1, 0,...



Source: https://arxiv.org/abs/1910.03193

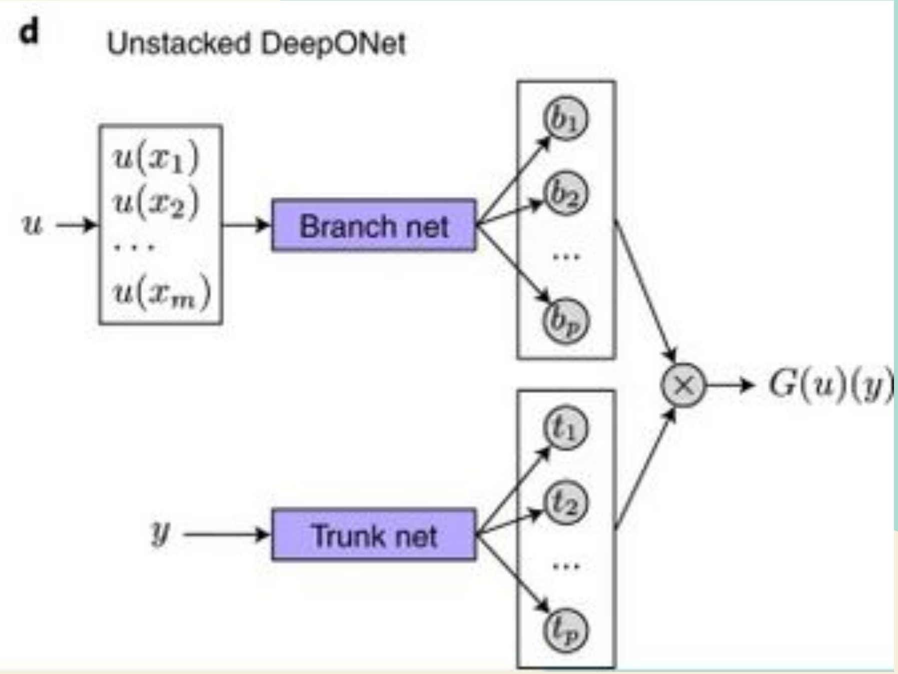
Universal Approximation Thm: Operator Version...

Theorem 1 (Universal Approximation Theorem for Operator). *Suppose that σ is a continuous non-polynomial function, X is a Banach Space, $K_1 \subset X$, $K_2 \subset \mathbb{R}^d$ are two compact sets in X and $\mathbb{R}^d$, respectively, V is a compact set in $C(K_1)$, G is a nonlinear continuous operator, which maps V into $C(K_2)$. Then for any $\epsilon > 0$, there are positive integers n, p, m, constants $c_i^k, \xi_{ij}^k, \theta_i^k, \zeta_k \in \mathbb{R}$, $w_k \in \mathbb{R}^d$, $x_j \in K_1$, $i = 1, \ldots, n$, $k = 1, \ldots, p$, $j = 1, \ldots, m$, such that*

$$\left| G(u)(y) - \sum_{k=1}^{p} \underbrace{\sum_{i=1}^{n} c_i^k \sigma\left(\sum_{j=1}^{m} \xi_{ij}^k u(x_j) + \theta_i^k \right)}_{branch} \underbrace{\sigma(w_k \cdot y + \zeta_k)}_{trunk} \right| < \epsilon \tag{1}$$

holds for all $u \in V$ and $y \in K_2$.

Lu, Jin & Karniadakis (2019)



Branch Network

$u(x_1, x_2 ... x_n) \rightarrow b(u) \in R^p$

Encoded input function mapped to a vector that represents output function

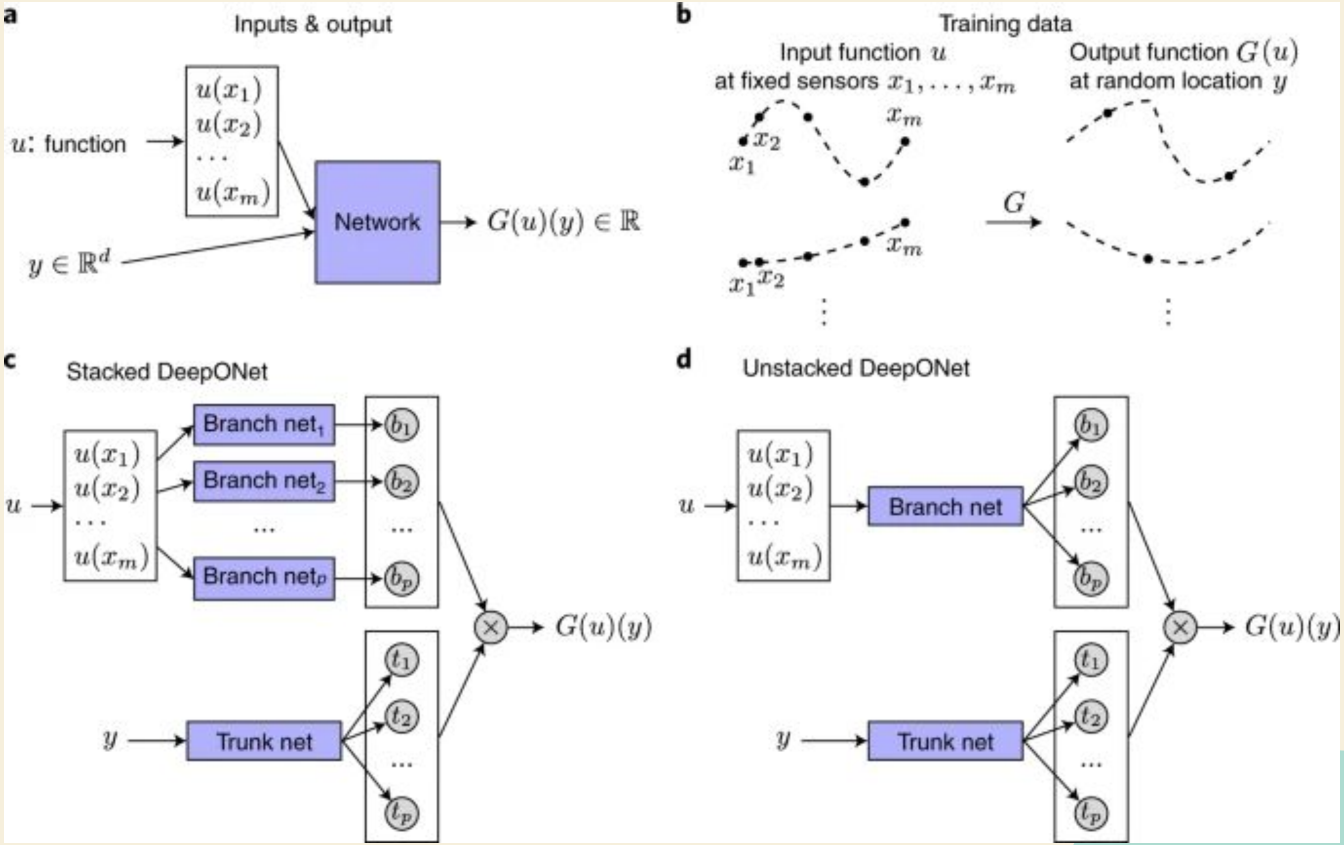
Trunk Network

$y \rightarrow t(y) \in R^p$

Input function inputs mapped to a vector

Final value: $G(u)(y) \cong b \cdot t$

...Inspires DeepONet



So what?

Conventional PDE solvers	Neural operators
Solve one instance	Learn a family of PDE
Require the explicit form	Black-box, data-driven
Speed-accuracy trade-off on resolution	Resolution-invariant, mesh-invariant
Slow on fine grids; fast on coarse grids	Slow to train; fast to evaluate

https://zongyi-li.github.io/neural-operator/

Applications

03

Example: Climate Modelling

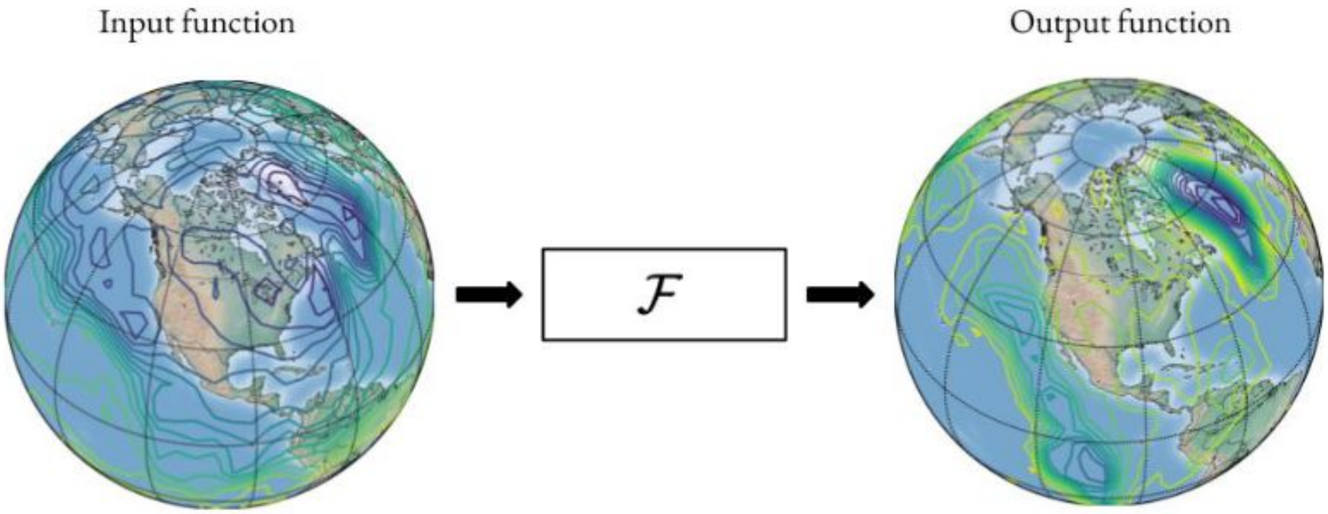
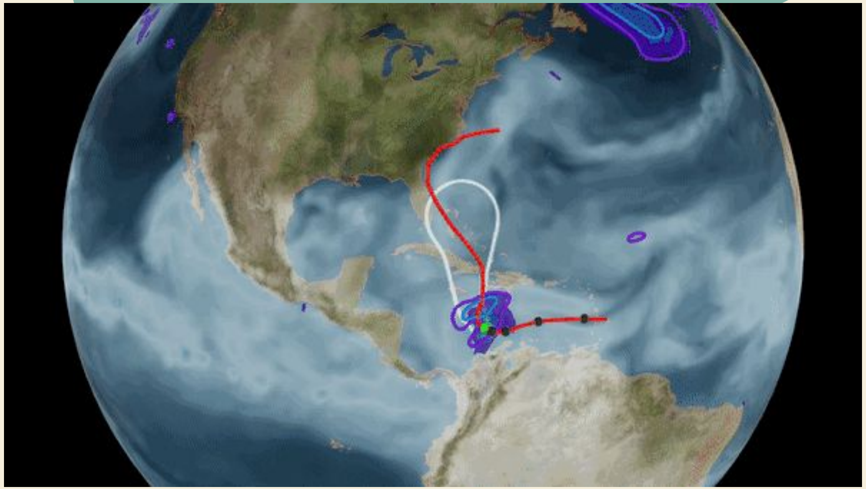


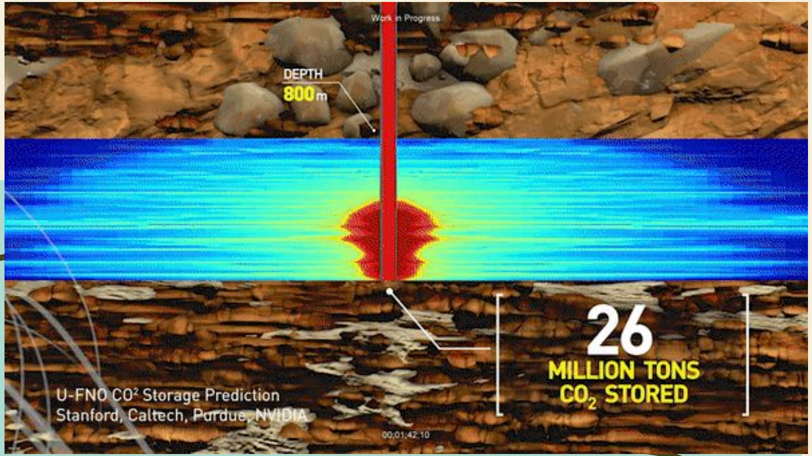
Figure 1: An example sketch of operator learning for climate modeling: By solving an operator learning problem, we can approximate an infinite-dimensional map between two functions of interest, and then predict one function using the other. For example, by providing the model with an input function, e.g. a surface air temperature field, we can predict an output function, e.g. the corresponding surface air pressure field.

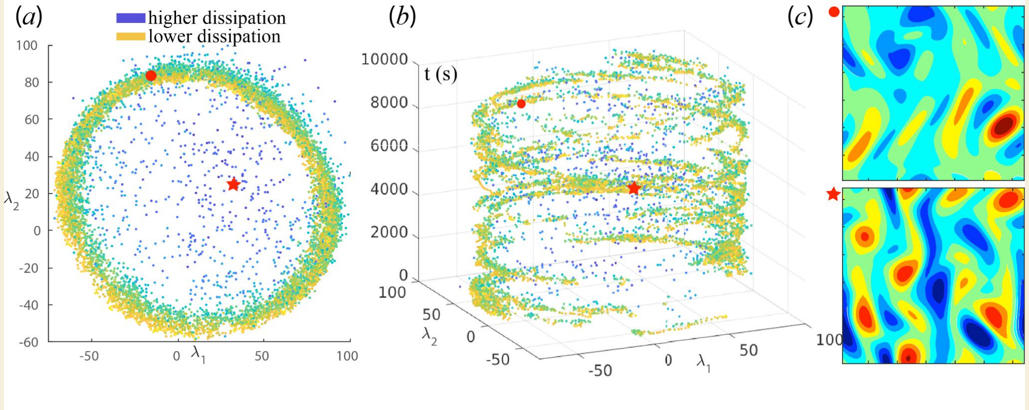
Kissas et. al. (2022).

Applications in:

- Hurricane predictions
- Chaotic systems (ie Kolmogorov Flow)
- Real-time calculations: Flight control
- Blood flow for medical imaging
- Carbon sequestration







https://zongyi-li.github.io/neural-operator/

Sources

Chen, Tianping, and Hong Chen. "Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems." *IEEE Transactions on Neural Networks* 6, no. 4 (1995): 911-917.

Karniadakis, George Em, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. "Physics-informed machine learning." *Nature Reviews Physics* 3, no. 6 (2021): 422-440.

Kissas, Georgios, Jacob H. Seidman, Leonardo Ferreira Guilhoto, Victor M. Preciado, George J. Pappas, and Paris Perdikaris. "Learning operators with coupled attention." *Journal of Machine Learning Research* 23, no. 215 (2022): 1-63.

Lu, Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators." *Nature Machine Intelligence* 3, no. 3 (2021): 218-229.

Seidman, Jacob H., Georgios Kissas, Paris Perdikaris, and George J. Pappas. "NOMAD: Nonlinear Manifold Decoders for Operator Learning." *arXiv preprint arXiv:2206.03551* (2022).

Questions?

Classical vs Operator ML

Classical	Operator
Approximates function f	Approximates operator L
f: vectors → vectors	L: functions → functions
Useful for: Finite-dimensional mapping	Useful for: Infinite-dimensional mapping; physical laws govern system
Value: easier to train	Value: simple forward-pass
Examples: images, NLP, product recommendation, (and more)	Examples: fluid flows, solid mechanics, climate modelling