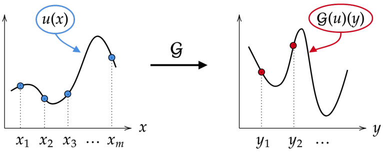
Deep Learning in Scientific Computing 2023: Lecture 11

Siddhartha Mishra

Seminar for Applied Mathematics (SAM), D-MATH (and),
ETH AI Center,
ETH Zürich, Switzerland.

Operator Learning

a $\mathcal{G}$a



- Underlying Solution Operator: $\mathcal{G}(a) = u$ for PDE $\mathcal{D}u = a$
- Task: Find a Surrogate (based on DNNs) $\mathcal{G}^* \approx \mathcal{G}$ from data.
- Inputs+Outputs for $\mathcal{G}^*$ are Functions.
- Some notion of Continuous-Discrete Equivalence

Operator Learning Architectures

$$
\begin{tikzcd}
X \arrow[r, "\mathcal{G}"] \arrow[d, squiggly, bend right, "\mathcal{E}"'] & Y \arrow[d, bend right, "\mathcal{P}"'] \\
\mathbb{R}^m \arrow[u, bend right, "\mathcal{D}"'] \arrow[r, squiggly, "\mathcal{A}"'] & \mathbb{R}^p \arrow[u, squiggly, bend right, "\mathcal{R}"']
\end{tikzcd}
$$

Architecture	Encoder	Approximator	Reconstructor
SNO[1]	Coeffs	DNNs	Chebychev basis
DeepOnet	Sensor Evals.	DNNs	DNNs
PCA-Net[2]	Input PCA	DNNs	Output PCA

[1] Fanaskov and Oseledets, 2022
[2] Bhattacharya et al, 2020

Alternative: Neural Operators

- Formalized in Kovachki et al, 2021.
- Recall: DNNs are $\mathcal{L}_\theta = \sigma_K \odot \sigma_{K-1} \odot \dots\dots \sigma_1$ u
- Single hidden layer: $\sigma_k(y) = \sigma(A_k y + B_k)$
- Neural Operators generalize DNNs to ∞-dimensions:
- NO: $\mathcal{N}_\theta = \mathcal{N}_L \odot \mathcal{N}_{L-1} \odot \dots\dots \mathcal{N}_1$ u — function
- Single hidden layer;

$$(\mathcal{N}_\ell v)(x) = \sigma \left(A_\ell v(x) + B_\ell(x) + \int_D K_\ell(x,y)v(y)dy \right)$$

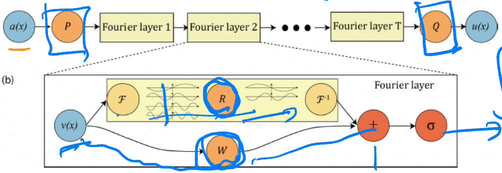
- Kernel Integral Operators
- Learning Parameters in A_ℓ, B_ℓ, K_ℓ
- Different Kernels $\Rightarrow$ Low-Rank NOs, Graph NOs, Multipole NOs, ...

Fourier Neural Operators

$G \approx G^*$

- FNO proposed in Li et al, 2020.

G^*:



$\int K(x,y)\, v(y)\, dy$

$B_\ell(x)$

- Translation invariant Kernel $\boxed{K(x,y) = K(x-y)}$
- Use Fourier and Inverse Fourier Transform to define the KIO:

$$\int_D K_\ell(x,y)v(y)dy = \boxed{\mathcal{F}^{-1}(\mathcal{F}(K)\mathcal{F}(v))(x)}$$

- Parametrize Kernel in Fourier space.
- Fast implementation through FFT

Theory for FNOs

— $\boxed{\text{Exactly like for DeepOnets}}$

$m_\epsilon \to \infty$
$\epsilon \to 0$

In general Size $(\mathcal{N}) \sim O\left(\epsilon^{-\frac{m_\epsilon}{k}}\right)$

m_ϵ
$= \#$ Sample points.

$k = 1$

- Results of Kovachki, Lanthaler,SM, 2021
- Universal Approximation Thm: For $\mu \in Prob(L^2(D))$ and any measurable $\mathcal{G}$: $H^r \mapsto H^s$ and $\epsilon > 0$, $\exists \mathcal{N}$ (FNO): $\hat{\mathcal{E}} < \epsilon$
- FNOs break the Curse of Dimensionality for a variety of PDEs.
- FNO Size grows polynomially wrt Error !

Size $(\mathcal{N}) \sim O\left(\epsilon^{-\frac{\alpha d}{k}}\right)$

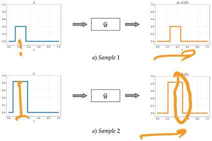
Continuous. $\mathcal{G} : L^2 \longmapsto L^2$

$$\hat{\mathcal{E}}^2 := \int \|\mathcal{G}(a) - \mathcal{G}^*(a)\|_{L^2}^2 \, d\mu(a)$$

DeepONet vs. FNO

- Theory of (Lanthaler, Molinaro, Hadorn, SM, 2022):
- For Linear Advection Equation with Discontinuities:

$u_0 \longmapsto u(\cdot, T)$



$u_t + a u_x = 0$

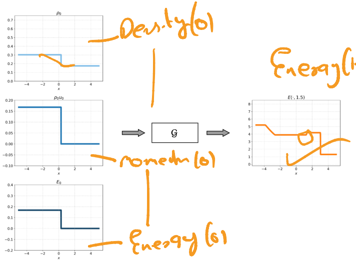
Recall: Eigs of Cov decay slowly:

- Thm: To obtain ϵ error:
  - Size(DeepOnet) $\sim \mathcal{O}(\epsilon^{-2})$ — Lower bound.
  - Size(FNO) $\sim \mathcal{O}(\log(\epsilon^{-1}))$!!
- Results:

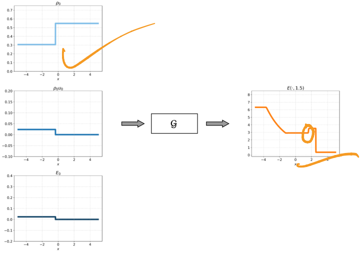
Architecture	ResNet	FCNN	DONet	FNO
Error	14.8%	23.3%	7.9%	0.7%

- Analogous theorem for Burgers' equation.

Operator Learning for Euler Equations



Sample 1



Sample 2

Architecture	Error
ResNet	4.5%
ConvNet	8.9%
DeepONet	4.2%
FNO	1.6%

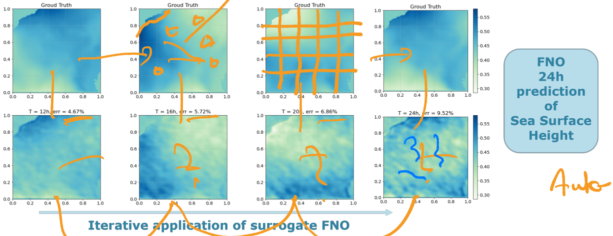
A more realistic problem: NEMO data set

Navier-Stokes Eqns

Ocean modelling

Sea Surface height

T=0 T=6 North Sea T=12 T=18

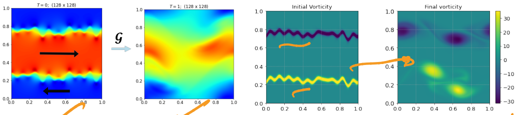


Auto Regression.

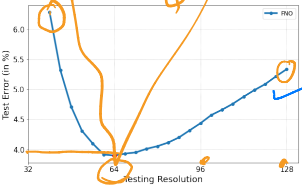
- Large errors with FNO.
- What is going on ?

Another Red Flag

2-D Navier-Stokes



- FNO Results:



- Recall Desiderata for Operator Learning:
  - Input + Output are functions.
  - Some notion of Resolution Invariance ?

A Possible Culprit: Aliasing Errors

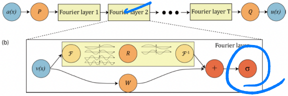
Destroy Continuous - Discrete Equivalence

$u(x) = \sum_j c_j \varphi_j(x)$

$u \mapsto \{c_j\}_{j=1}^{J}$

$\varphi_j \to$ Fourier basis.

- We evaluate FNO inputs on a grid:



Poisson Summation formula

- Can lead to Aliasing: mismatch between continuous and discrete

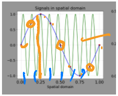
Point values

Averages

"Coefficients of a basis"

digitize functions:
- discretize functions

Nyquist Barrier

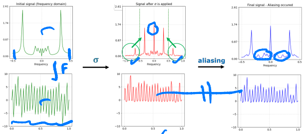


Point value

No mismatch between continuous and discrete

Shannon Sampling Theorem

- Particularly due to the Activation:



$\int F$

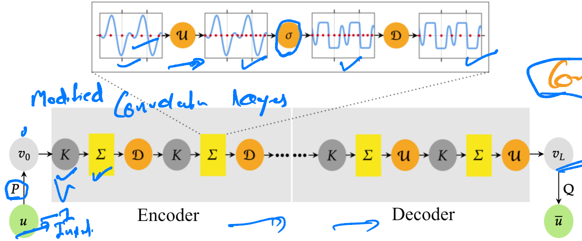
u $\sigma(u)$

$\sin(x) = \cos(u) -$

Convolutions Strike Back !!

- Convolutional Neural Operators (CNOs) of Raonic et al, 2023.

Base on Convolutions. (locality)



- Alias-Free as consistent with Sampling Theorems
- Structure Preserving Representation Equivalent Neural Operator or ReNO
  - continuous - discrete Equivalence
- Universal for $\mathcal{G} : H^r \mapsto H^s$

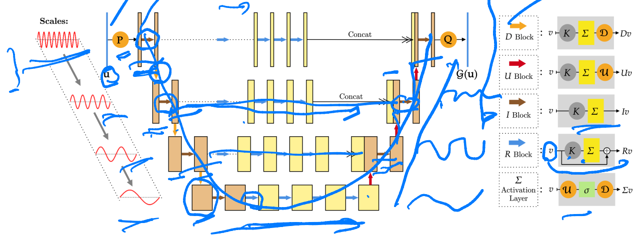
Approximation $L^2 \longmapsto L^2$ Exactly like FNO, DeepOnet

CNO Architecture in Practice

Operator U-Nets

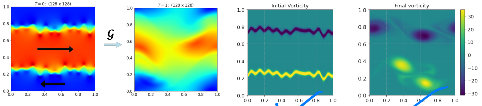
Lifting

Downsampling

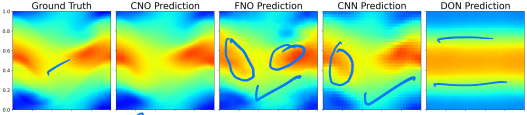


Results for Navier-Stokes Shear Flow

- Operator:



- Comparison:



- Resolution Dependence:

Results for Poisson

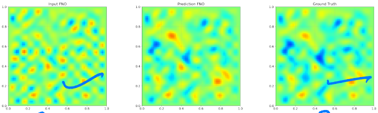
$-\Delta u = f$

$\boxed{f = \sum_{i,j=1}^{16} a_{ij} \sin(i\pi x) \sin(j\pi y)}$

$a_{ij} \sim$ Unif $[-1,1]$

- FNO Approximation:

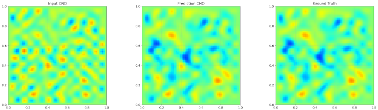
$\{a_{ij}\}_{i,j=1}^{16} \longmapsto u$



Analytical Exact Soln

16 → 20

- CNO Approximation:



f   u

- In-Distribution Test Errors: 4.8% (FNO) vs 0.23% (CNO) vs 12.9% (DeepOnet)
- Out-of-Distribution Test Errors: 8.9% (FNO) vs 0.27% (CNO) vs 9.2% (DeepOnet)

Testing.

Results for Wave Equation

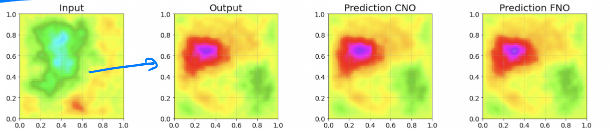
– Standing wave

$u_{tt} - c^2 \Delta u = 0$

$u(t=0) = u_0$

$u_t(t=0) = u_1$

- $\mathcal{G} : u_0 \mapsto u(T)$

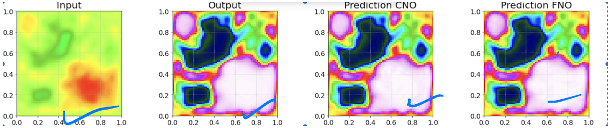


- In-Distribution Test Errors: 1.1% (FNO) vs 0.83% (CNO) vs 2.3% (DeepOnet)
- Out-of-Distribution Test Errors: 1.6% (FNO) vs 1.48% (CNO) vs 2.9% (DeepOnet)

Results for Allen-Cahn Equation

$u_t = \epsilon \Delta u - \epsilon u(u^2 - 1)$

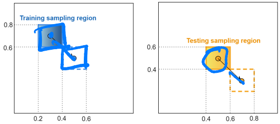
- $\mathcal{G} : u_0 \mapsto u(T)$



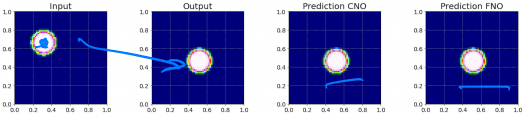
- In-Distribution Test Errors: 0.6% (FNO) vs 0.8% (CNO) vs 13.6% (DeepOnet)
- Out-of-Distribution Test Errors: 2.4% (FNO) vs 3.6% (CNO) vs 19.9% (DeepOnet)

Results for Linear Transport Equation: I

- In-Distribution testing:



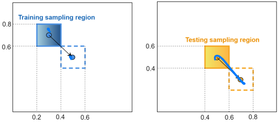
- $\mathcal{G} : u_0 \mapsto u(T)$



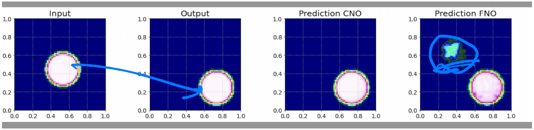
- Errors: 1.29% with FNO vs 1.17% with CNO vs 5.78% with DeepOnet

Results for Linear Transport Equation: II

- Out-of-Distribution testing:



- $\mathcal{G} : u_0 \mapsto u(T)$



- Errors: 8.78% with FNO vs 1.6% with CNO vs 117.1% with DeepOnet

Compressible Euler Equations

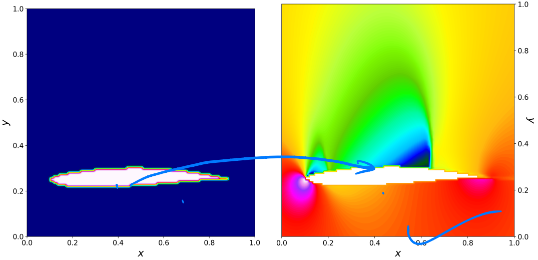
O(100s)

- Operator maps shape to Density.

30

20

Hicks-Henne functions.



1.5/1.6

- In-Distribution Test Errors: 0.47% (FNO) vs 0.35% (CNO) vs 1.93% (DeepOnet)
- Out-of-Distribution Test Errors: 0.85% (FNO) vs 0.62% (CNO) vs 2.88% (DeepOnet)

Operator Learning: Open Issues

$\mathcal{G}^*: a \mapsto u^*$ $\quad \mathcal{G}^* a$

$\mathcal{D}(\mathcal{G}^* a) = a$

- Scale to Realistic 3D problems
- Predict Long Time Series with Autoregression
- Limited Data $\Rightarrow$ use PDE losses
  - Combine operator learning with PINNs
  - Leads to PI-Onets, PINOs, PICNOs.
- Out of Distribution Generalization
- Non-trivial extension of PDE Inverse Problems

$O(10)$

Bayesian Inverse Problems

UQ / Optimization