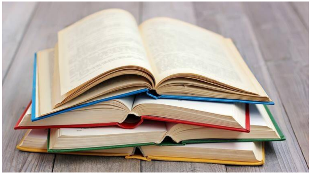
Discretization-Invariant Operator Learning: Algorithms and Theory

Haizhao Yang
Department of Mathematics
Purdue University

Workshop of Scientific Computing meets Machine Learning and Life Sciences
Texas Tech University
March 5th, 2022

Conventional Solvers vs. Data-Driven Methods

New diagram for solutions and new opportunities for mathematics



Conventional solvers

- Years of design to solve
- Months of coding
- Accurate but maybe slow

Data-driven methods

- Learning to solve from data
- Days or months of training
- Fair and fast solution

Learning Mathematical Operators

Notations

- Function spaces $\mathcal{X}$ and $\mathcal{Y}$, e.g., $\mathbb{R}$-valued over domain $\Omega \subset \mathbb{R}^D$
- Operator $\Psi : \mathcal{X} \to \mathcal{Y}$
- Data samples $\mathcal{S} = \{u_i, v_i\}_{i=1}^{2n}$ with

$$v_i = \Psi(u_i) + \epsilon_i,$$

  where $u_i \overset{\text{i.i.d.}}{\sim} \gamma$ and $\epsilon_i \overset{\text{i.i.d.}}{\sim} \mu$

Goal

- Learn Ψ from samples $\mathcal{S}$

Method

- Deep neural networks $\Psi^n(u;\theta)$ as parametrization
- Supervised learning to find $\Psi^n(\cdot;\theta^*) \approx \Psi(\cdot)$

Why Operator Learning?

Broad applications

- Reduced order modeling: learning operators in lower dim
- Solving parametric PDEs
- Solving inverse problems
- Density function theory: potential function to density function
- Phase retrieval: data to images
- Image processing: image to image
- Predictive data science: historical states to future states

Probably most mappings are high-dim or even infinite-dim

Why Discretization-Invariant

Main concern in applications

- Given accuracy, minimize cost

Key difficulties

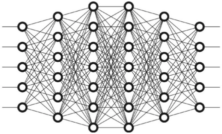
- A nonlinear operator Ψ between infinite-dimensional $\mathcal{X}$ and $\mathcal{Y}$
- Heterogeneous data structures

Part I: Operator Learning Algorithm

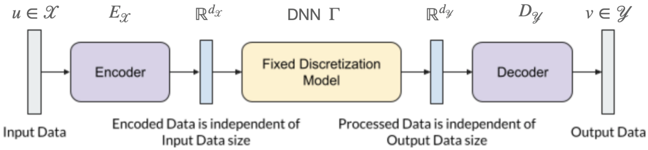
Deep neural network

$$v = \Psi(u;\theta) := T \circ h^{(L)} \circ h^{(L-1)} \circ \cdots \circ h^{(1)}(u)$$

- $h^{(i)}(u) = \sigma({W^{(i)}}^T u + b^{(i)})$
- Activation function $\sigma(x)$, e.g. $\mathrm{ReLU}(x) = \max\{0, x\}$
- $T(v) = V^T v$
- $\theta = (W^{(1)}, \cdots, W^{(L)}, b^{(1)}, \cdots, b^{(L)}, V)$

$u \in \mathbb{R}^{d_x} \rightarrow$  $\rightarrow v \in \mathbb{R}^{d_y}$

Operator Learning with **Fixed** Input and Output Sizes



Most methods:

Encoder-decoder of $\mathcal{X}$

- $D_\mathcal{Y} \circ E_\mathcal{X} \approx I$, $E_\mathcal{X} : \mathcal{X} \to \mathbb{R}^{d_\mathcal{X}}$, $D_\mathcal{Y} : \mathbb{R}^{d_\mathcal{X}} \to cX$
- Encoder $E_\mathcal{X}$: sampling, basis expansion, PCA, etc.
- Decoder $D_\mathcal{X}$: interpolation, basis expansion, PCA, etc.

Encoder-decoder of $\mathcal{Y}$

- Similar

Learning

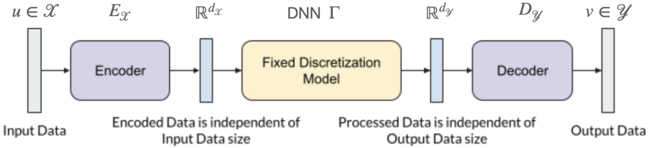
- A DNN $\Gamma \approx \bar{\Psi} : \mathbb{R}^{d_\mathcal{X}} \to \mathbb{R}^{d_\mathcal{Y}}$
- $D_\mathcal{Y} \circ \Gamma \circ E_\mathcal{X} \approx \Psi : \mathcal{X} \to \mathcal{Y}$

Repeated and expensive re-training if $d_\mathcal{X}$ or $d_\mathcal{Y}$ changes

Discretization Invariant Operator Learning

Ong, Shen, Y., preprint, 2022
Sparsity: Key to discretization-invariance

Our idea 1 of network construction



Encoder and decoder

- Discretization-invariant
- Capture intrinsic dimension (sparsity)

Fixed discretization model

- Powerful expressivity
- Deep neural network (DNN)

Discretization Invariant Operator Learning

Ong, Shen, Y., preprint, 2022

Our integral-kernel-based encoder

$$v(y) = \int_{\Omega_{\mathcal{X}}} \phi_1(x, y; \theta_1) u(x) dx$$

- Mapping $u \in \mathcal{X}$ to $v(y) \in \mathcal{Y}$ defined for $y \in \Omega_{\mathcal{Y}}$
- Kernel ϕ_1 is a DNN parametrized by θ_1
- $\int_{\Omega_{\mathcal{X}}}$ is discretized according to the discrete $u(x)$

Our integral-kernel-based decoder

$$u(x) = \int_{\Omega_{\mathcal{Y}}} \phi_2(x, y; \theta_2) v(y) dy$$

- Mapping $v \in \mathcal{Y}$ to $u(x) \in \mathcal{X}$ defined for $x \in \Omega_{\mathcal{X}}$
- Kernel ϕ_2 is a DNN parametrized by θ_2
- $\int_{\Omega_{\mathcal{Y}}}$ is discretized according to the discrete $v(y)$

Discretization Invariant Operator Learning

Ong, Shen, Y., preprint, 2022

Why integral-kernel-based encoder and decoder?

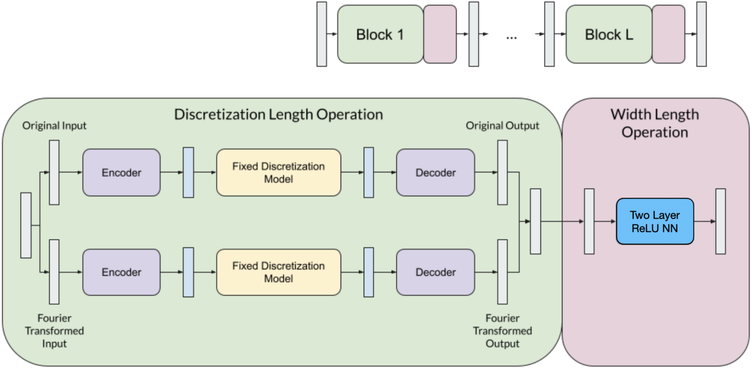
$$v(y) = \int_{\Omega} \phi(x, y; \theta) u(x) dx$$

- DNN expressivity: Fourier, Wavelet, other integral operators
- Data driven sparsity, i.e., DNN-based PCA

Discretization Invariant Operator Learning

Our idea 2 of network construction

- Parallel blocks (e.g., spatial and frequency domains)
- Post-processing ReLU NN
- Deep network via densely connected composition



Discretization Invariant Operator Learning

Our idea 3 for randomized data augmentation

Loss function

$$\min_{\theta} \mathbb{E}_{(u,v)\sim p_{data}} \mathbb{E}_S \left[\mathcal{L} \left(\Psi(u;\theta), v \right) + \lambda \mathcal{L} \left(\Psi(S(u);\theta), S(v) \right) \right]$$

- $\Psi(u;\theta)$ discretization-invariant neural network
- $\mathcal{L}(\cdot,\cdot)$: typical loss function, e.g., $\mathcal{L}(x,y) = ||x-y||^2$
- Random interpolation operator S
- p_{data}: joint distribution of (u,v) in $\mathcal{X} \times \mathcal{Y}$
- $\lambda > 0$

Numerical Comparison

Existing methods

- **UNet**, Ronneberger et al., MICCAI, 2015
- **DeepOnet**, Lu et al., Nature Machine Intelligence, 2021
- **FNO** (Fourier Neural Operator), Li et al., ICLR 2021
- **FT** (Fourier Transformer) and **GT** (Galerkin Transformer), S. Cao, NeurIPS, 2021

Examples

- Prediction
- Forward problems
- Inverse problems
- Signal processing

Numerical Comparison

Prediction of future states

Example 1: Burgers equation:

$$\partial_t u(x,t) + \partial_x(u^2(x,t)/2) = \nu \partial_{xx} u(x,t), \quad x \in (0,1), t \in (0,1]$$
$$u(x,0) = u_0(x)$$

- Periodic boundary conditions
- $\nu = 0.1$: a given viscosity coefficient
- Applications in fluid mechanics, nonlinear acoustics, gas dynamics, and traffic flow
- Goal: learn the mapping from $u_0(x)$ to $u(x,1)$.

Numerical Comparison

Example 1: Burgers equation:

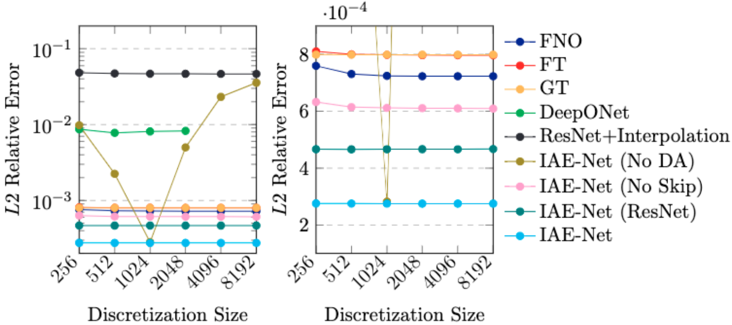


Figure: $L2$ relative error with $\nu = 1e^{-1}$ (Left) and its closeup (Right). Models are trained with $s = 1024$ and tested on the other resolutions.

Numerical Comparison

Example 1: Burgers equation:

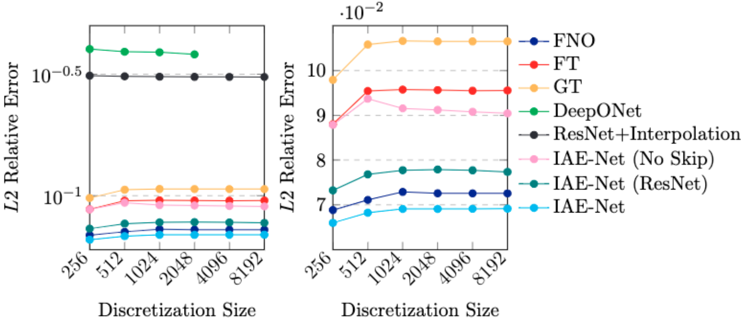
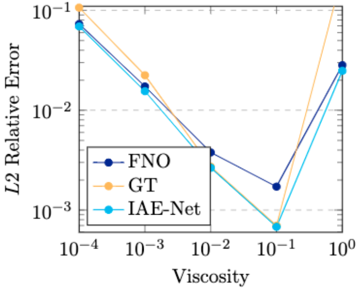


Figure: $L2$ relative error with $\nu = 1e^{-4}$ (Left) and its closeup (Right). Models are trained with $s = 1024$ and tested on the other resolutions.

Numerical Comparison

Example 1: Burgers equation:



(a) Comparison of relative error for burgers equation with varying ν.

Numerical Comparison

Forward problem

Example 2: the steady-state of the 2D Darcy Flow equation:

$$-\nabla \cdot (a(x)\nabla u(x)) = f(x), \quad x \in (0,1)^2$$
$$u(x) = 0, \quad x \in \partial(0,1)^2$$

- f: a given forcing function
- Applications in modeling the pressure of subsurface flow, the deformation and the electric potential of materials
- Goal: learn the forward mapping from $a(x)$ to $u(x)$.

Numerical Comparison

Example 2: the steady-state of the 2D Darcy Flow equation:

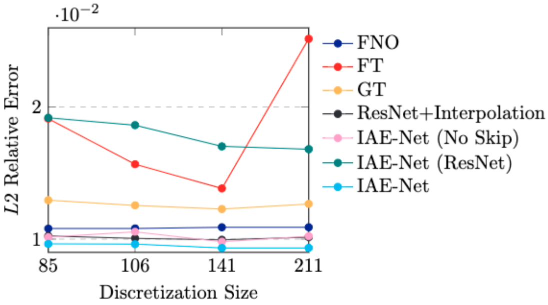


Figure: $L2$ relative error. Models are trained with $s = 141$ size training data and tested on the other resolutions.

Numerical Comparison

Inverse problem

Example 3: inverse scattering.

- Applications: non-destructive testing, medical imaging, seismic imaging, etc.
- Helmholtz equation

$$\left(-\nabla - \frac{\omega^2}{c(x)^2}\right) u(x) = 0$$

with a given frequency ω and unknown speed $c(x)$

- Introduce

$$\frac{\omega^2}{c(x)^2} = \frac{\omega^2}{c_0(x)^2} + \eta(x), \qquad L_0 = -\nabla - \frac{\omega^2}{c_0(x)^2}$$

with $c_0(x)$ given in applications

- Helmholtz equation:

$$\left(-\nabla - \frac{\omega^2}{c(x)^2}\right) u(x) = (L_0 - \eta(x))u(x) = 0$$

as a parametric PDE with parameter η

- Goal: learn the mapping from $u(x)$ at sensor locations to $\eta(x)$

Numerical Comparison

Inverse problem

Example 3: inverse scattering.

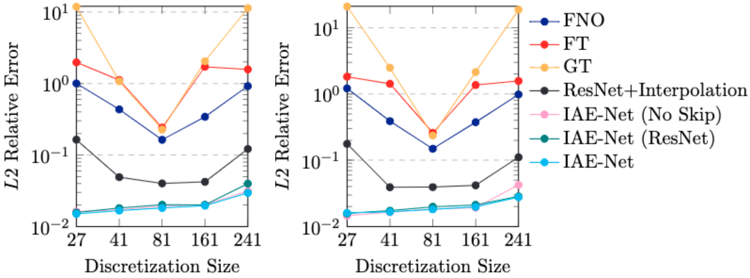


Figure: $L2$ relative error for the forward (Left) and inverse (Right) problem. Model is trained with $s = 81$ and tested on different resolutions.

Numerical Comparison

Image/signal processing

Example 4: blind source separation.

- Applications in image processing, medical imaging, audio signal, health measurement

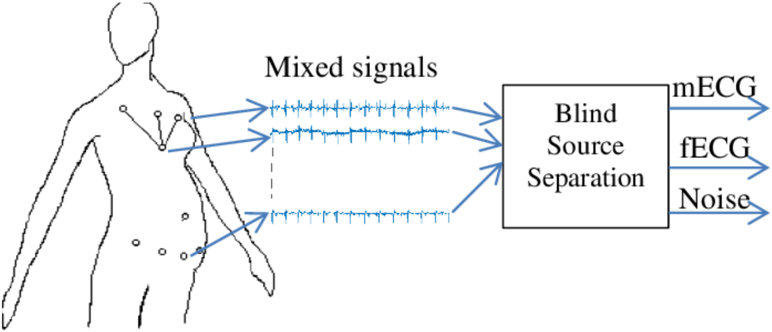


Figure: Extracting fetal ECG from mother's measurement plays an important role in diagnosing fetus's health. Figure credited to Bensafia et al.

Numerical Comparison

Example 4: blind source separation.

Table: Trained with size $s = 2000$ and tested on different resolutions for zero-shot generalization.

Model Name	**250**	**500**	**1000**	**2000**	**4000**
FNO	45.07%	24.75%	16.76%	15.97%	18.23%
GT	45.30%	27.24%	18.97%	17.75%	19.2%
DeepONet†				99.99%	
Unet	112%	101%	68.78%	8.274%	69.85%
ResNet + *Interpolation*	66.37%	43.73%	32.13%	31.16%	31.92%
IAE-Net (No Skip)	15.36%	10.68%	8.723%	7.904%	8.153%
IAE-Net (ResNet)	14.08%	9.924%	7.925%	7.15%	7.192%
IAE-Net	12.08%	8.638%	7.048%	6.802%	6.848%

Part II: Operator Learning theory

Literature

Existing theory

- A posteriori error analysis for DeepOnet[1]
- Non-DNN approach for linear operators[2]

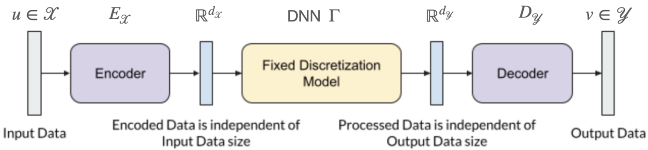
Our goal

- A priori error analysis
- Nonlinear operators
- Discretization-invariant

[1] S. Lanthaler, S. Mishra, and G. E. Karniadakis. Error estimates for deepOnets: A deep learning framework in infinite dimensions. arXiv:2102.09618, 2021.

[2] M. V. de Hoop, N. B. Kovachki, N. H. Nelsen, and A. M. Stuart. Convergence rates for learning linear operators from noisy data. arXiv:2108.12515, 2021.

An Abstract Basic Framework



Encoder-decoder

- Most methods $\mathcal{X} \approx \mathbb{R}^{d_\mathcal{X}} \to \mathbb{R}^{d_\mathcal{Y}} \approx \mathcal{Y}$; finite basis expansion
- PCA-Net[3] $\mathcal{X} \approx \mathbb{R}^{d_\mathcal{X}} \to \mathbb{R}^{d_\mathcal{Y}} \approx \mathcal{Y}$; PCA
- DeepOnet $\mathcal{X} \approx \mathbb{R}^{d_\mathcal{X}} \to \mathcal{Y}$; $E_\mathcal{X}$: function sampling; $D_\mathcal{Y}$: DNN basis functions
- Our algorithm with one block, $\mathcal{X} \to \mathcal{Y}$; DNN kernels

[3] Bhattacharya, Hosseini, Kovachki, Stuart, 2019

Problem Statement

Learning $\Psi \approx D_{\mathcal{Y}} \circ \Gamma \circ E_{\mathcal{X}}$

- Target Lip. operator $\Psi : \mathcal{X} \to \mathcal{Y}$
- Samples $\mathcal{S} = \{u_i, v_i\}_{i=1}^{2n}$ with $v_i = \Psi(u_i) + \epsilon_i$, $u_i \overset{\text{i.i.d.}}{\sim} \gamma$, and $\epsilon_i \overset{\text{i.i.d.}}{\sim} \mu$
- Step 1: use $\{u_i, v_i\}_{i=1}^{n}$ to learn encoder-decoder s.t.

$$D_{\mathcal{X}} \circ E_{\mathcal{X}} \approx I \quad \text{and} \quad D_{\mathcal{Y}} \circ E_{\mathcal{Y}} \approx I$$

- Step 2: use $\{u_i, v_i\}_{i=n+1}^{2n}$ to learn DNN Γ_θ via empirical risk

$$\min_{\Gamma_\theta \in \mathcal{F}_{\mathrm{NN}}} R_S(\theta) := \min_{\Gamma_\theta \in \mathcal{F}_{\mathrm{NN}}} \frac{1}{n} \sum_{i=n+1}^{2n} \|D_{\mathcal{Y}} \circ \Gamma_\theta \circ E_{\mathcal{X}}(u_i) - v_i\|_{\mathcal{Y}}^2$$

- Population risk (accuracy) of $\Psi_\theta := D_{\mathcal{Y}} \circ \Gamma_\theta \circ E_{\mathcal{X}} \approx \Psi$

$$R_D(\theta) := \mathbb{E}_{\mathcal{S}} \mathbb{E}_{u \sim \gamma} \|\Psi_\theta(u) - \Psi(u)\|_{\mathcal{Y}}^2$$

Question: How good is the empirical solution Ψ_{θ^*} with $\theta^* \in \operatorname{argmin} R_S(\theta)$?

Problem Statement

The goal of error analysis

Quantify

$$R_D(\theta^*) := \mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|\Psi_{\theta^*}(u) - \Psi(u)\|_{\mathcal{Y}}^2$$

in terms of DNN width, depth, and $\#$samples

Key questions

- Practical guidance on the choice of DNNs and samples
- Curse of dimensionality (in $\#$parameters and $\#$samples)
- Zero/few-shot generalization for different data structures

Problem Statement

Error analysis of $R_D(\theta^*)$

- Error decomposition

$$\begin{aligned} R_D(\theta^*) &= \mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\|D_{\mathcal{Y}} \circ \Gamma_{\theta^*} \circ E_{\mathcal{X}}(u) - \Psi(u)\|_{\mathcal{Y}}^2\right] \\ &= T_1 + T_2 \end{aligned}$$

- Bias (approximation)

$$T_1 = 2\mathbb{E}_{\mathcal{S}}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\|D_{\mathcal{Y}} \circ \Gamma_{\theta^*} \circ E_{\mathcal{X}}(u_i) - \Psi(u_i)\|_2^2\right]$$

- Variance (generalization)

$$\begin{aligned} T_2 = {} & \mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\|D_{\mathcal{Y}} \circ \Gamma_{\theta^*} \circ E_{\mathcal{X}}(u) - \Psi(u)\|_{\mathcal{Y}}^2\right] \\ & - 2\mathbb{E}_{\mathcal{S}}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\|D_{\mathcal{Y}} \circ \Gamma_{\theta^*} \circ E_{\mathcal{X}}(u_i) - \Psi(u_i)\|_2^2\right] \end{aligned}$$

First step: estimation of T_1 via DNN approximation

$$T_1 = 2\mathbb{E}_{\mathcal{S}}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\|D_{\mathcal{Y}} \circ \Gamma_{\theta^*} \circ E_{\mathcal{X}}(u_i) - \Psi(u_i)\|_2^2\right]$$

Our goals in approximation

- Approximation error in terms of width and depth
- Does curse of dim (e.g., $\#$ parameters $(\frac{1}{\epsilon})^d$) exist?

Literature Review

Active research directions

Cybenko, 1989; Hornik et al., 1989; Barron, 1993; Montufar, Ay, 2011; Liang and Srikant, 2016; Yarotsky, 2017; Poggio et al., 2017; Schmidt-Hieber, 2017; E and Wang, 2018; Petersen and Voigtlaender, 2018; Chui et al., 2018; Yarotsky, 2018; Nakada and Imaizumi, 2019; Gribonval et al., 2019; Gühring et al., 2019; Chen et al., 2019; Li et al., 2019; Suzuki, 2019; Bao et al., 2019; E et al., 2019; Opschoor et al., 2019; Merkh, Montufar, 2019; Yarotsky and Zhevnerchuk, 2019; Bölcskei et al., 2019; Montanelli and Du, 2019; Chen and Wu, 2019; Zhou, 2020; Montanelli et al., 2020, etc.

ReLU DNNs, continuous functions $C([0,1]^d)$

ReLU; Fixed network width $O(N)$ and depth $O(L)$

- Nearly tight error rate $5\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d})$ simultaneously in N and L with L^∞-norm. Shen, Y., and Zhang (CiCP, 2020)
- ω_f is the modulas of continuity
- Improved to a tight rate $O\left(\sqrt{d}\,\omega_f\left(\left(N^2L^2\log_3(N+2)\right)^{-1/d}\right)\right)$. Shen, Y., and Zhang (J Math Pures Appl, 2021)

Remark

- Curse of dim exists
- Smoothness cannot help (Lu, Shen, Y., Zhang, SIMA, 2021)
- Need special function structures or activation functions to lessen the curse

Second step: estimation of T_2 via DNN generalization

$$
\begin{aligned}
T_2 = {} & \mathbb{E}_{\mathcal{S}} \mathbb{E}_{u \sim \gamma} \left[\| D_{\mathcal{Y}} \circ \Gamma_{\theta^*} \circ E_{\mathcal{X}}(u) - \Psi(u) \|_{\mathcal{Y}}^2 \right] \\
& - 2 \mathbb{E}_{\mathcal{S}} \left[\frac{1}{n} \sum_{i=n+1}^{2n} \| D_{\mathcal{Y}} \circ \Gamma_{\theta^*} \circ E_{\mathcal{X}}(u_i) - \Psi(u_i) \|_2^2 \right]
\end{aligned}
$$

Deep Network Generalization

Active research directions

Hamers and Kohler 2006; Jacot, Gabriel, and Hongler 2018; Bauer and Kohler 2019; Cao and Gu 2019; Chen et al. 2019; Schmidt-Hieber 2020; Kohler, Krzyzak, and Langer 2020; Nakada and Imaizumi 2020; Farrell, Liang, and Misra 2021; Jiao, Shen, Lin, and Huang 2021, etc

Remark

Very limited for operator learning

Deep Network Generalization

Road map (Liu, Y.*, Chen, Zhao, Liao*, arXiv:2201.00217, 2022)

- Variance $T_2 \to$ covering number of $\mathcal{F}_{\mathrm{NN}}$
- Covering number of $\mathcal{F}_{\mathrm{NN}} \to$ pseudo-dimension of $\mathcal{F}_{\mathrm{NN}}$
- Pseudo-dimension of $\mathcal{F}_{\mathrm{NN}} \to$ NN width and depth

Full Error Analysis

Theorem ((Liu, Y.*, Chen, Zhao, Liao*, arXiv:2201.00217))

Under certain assumptions. Let Γ_{θ^*} *be the minimizer of the empirical loss with depth* $L = O(\widetilde{L}\log\widetilde{L})$*, width* $N = O\left(\widetilde{p}\log\widetilde{p}\right)$*, magnitude bound* $M = O(\sqrt{d_{\mathcal{Y}}})$*, where* $\widetilde{L}, \widetilde{p}$ *are positive integers satisfying*

$$\widetilde{L}\widetilde{p} = \left\lceil d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{4+2d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{4+2d_{\mathcal{X}}}} \right\rceil .$$

Then we have

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}}\circ\Gamma_{\theta^*}\circ E_{\mathcal{X}}(u)-\Psi(u)\|_{\mathcal{Y}}^2 \\
\leq\, & O\left((\sigma^2+1)d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}}\log^6 n\right) \\
&+ O\left(\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{X}}\circ E_{\mathcal{X}}(u)-u\|_{\mathcal{X}}^2 + \mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\|D_{\mathcal{Y}}\circ E_{\mathcal{Y}}(w)-w\|_{\mathcal{Y}}^2\right)
\end{aligned}$$

Interpretation

- Curse of dim exists
- Require accurate encoding for zero/few-shot generalization

Additional Low-Dimensional Structures

Assumption (low-dimensional manifold)
$\{E_{\mathcal{X}}(u) : u \sim \gamma\}$ lie on a d_0-dimensional manifold with $d_0 \ll d_{\mathcal{X}}$

Theorem ((Liu, Y.*, Chen, Zhao, Liao*, arXiv:2201.00217))
In addition to the above assumption, we have

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}} \circ \Gamma_{\theta^*} \circ E_{\mathcal{X}}(u) - \Psi(u)\|_{\mathcal{Y}}^2 \\
\leq\; &O\left((\sigma^2 + 1) d_{\mathcal{Y}}^{\frac{4+d_0}{2+d_0}} n^{-\frac{2}{2+d_0}} \log^6 n\right) \\
&\quad + O\left(\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{X}} \circ E_{\mathcal{X}}(u) - u\|_{\mathcal{X}}^2 + \mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\|D_{\mathcal{Y}} \circ E_{\mathcal{Y}}(w) - w\|_{\mathcal{Y}}^2\right)
\end{aligned}
$$

Additional Low-Dimensional Structures

Assumption (low complexity)

$$D_{\mathcal{Y}} \circ E_{\mathcal{Y}} \circ \Psi(u) = D_{\mathcal{Y}} \circ \mathbf{g} \circ E_{\mathcal{X}}(u)$$

with $\mathbf{g} : \mathbb{R}^{d_{\mathcal{X}}} \to \mathbb{R}^{d_{\mathcal{X}}}$ in the form:

$$\mathbf{g}(\mathbf{a}) = \begin{bmatrix} g_1(V_1^\top \mathbf{a}) & \cdots & g_{d_{\mathcal{Y}}}(V_{d_{\mathcal{Y}}}^\top \mathbf{a}) \end{bmatrix}^\top,$$

for $V_k \in \mathbb{R}^{d_{\mathcal{X}} \times d_0}$, and $g_k : \mathbb{R}^{d_0} \to \mathbb{R}$ (multi-index models).

Theorem ((Liu, Y.*, Chen, Zhao, Liao*, arXiv:2201.00217))

In addition to the above assumption, we have

$$\begin{aligned} &\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}} \circ \Gamma_{\theta^*} \circ E_{\mathcal{X}}(u) - \Psi(u)\|_{\mathcal{Y}}^2 \\ \leq\ & O\left((\sigma^2+1) d_{\mathcal{Y}}^{\frac{4+d_0}{2+d_0}} \max\left\{n^{-\frac{2}{2+d_0}}, d_{\mathcal{X}} n^{-\frac{4+d_0}{4+2d_0}}\right\} \log^6 n\right) \\ &+ O\left(\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{X}} \circ E_{\mathcal{X}}(u) - u\|_{\mathcal{X}}^2 + \mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\|D_{\mathcal{Y}} \circ E_{\mathcal{Y}}(w) - w\|_{\mathcal{Y}}^2\right) \end{aligned}$$

Acknowledgment

Collaborators

Minshuo Chen, Wenjing Liao, Hao Liu, Yong Zheng Ong, Zuowei Shen, Tuo Zhao

Funding

NSF

ORACLE LABS

ExxonMobil