Journal of Machine Learning Research 25 (2024) 1-67 Submitted 6/22; Revised 12/23; Published 1/24

Deep Nonparametric Estimation of Operators between Infinite Dimensional Spaces

Hao Liu[1] HAOLIU@HKBU.EDU.HK
Haizhao Yang[2*] HZYANG@UMD.EDU
Minshuo Chen[3] MINSHUOCHEN@PRINCETON.EDU
Tuo Zhao[4] TOURZHAO@GATECH.EDU
Wenjing Liao[5*] WLIAO60@GATECH.EDU

[*] *Co-corresponding author*

[1] *Department of Mathematics, Hong Kong Baptist University, Hong Kong*

[2] *Department of Mathematics and Department of Computer Science, University of Maryland College Park, USA*

[3] *Department of Electrical and Computer Engineering, Princeton University, USA*

[4] *School of Industrial and Systems Engineering, Georgia Institute of Technology, USA*

[5] *School of Mathematics, Georgia Institute of Technology, USA*

Editor: Maxim Raginsky

Abstract

Learning operators between infinitely dimensional spaces is an important learning task arising in machine learning, imaging science, mathematical modeling and simulations, etc. This paper studies the nonparametric estimation of Lipschitz operators using deep neural networks. Non-asymptotic upper bounds are derived for the generalization error of the empirical risk minimizer over a properly chosen network class. Under the assumption that the target operator exhibits a low dimensional structure, our error bounds decay as the training sample size increases, with an attractive fast rate depending on the intrinsic dimension in our estimation. Our assumptions cover most scenarios in real applications and our results give rise to fast rates by exploiting low dimensional structures of data in operator estimation. We also investigate the influence of network structures (e.g., network width, depth, and sparsity) on the generalization error of the neural network estimator and propose a general suggestion on the choice of network structures to maximize the learning efficiency quantitatively.

Keywords: Deep neural networks, Nonparametric estimation, Operator learning, Generalization error analysis

1. Introduction

Learning nonlinear operators from a Hilbert space to another via nonparametric estimation has been an important topic with broad applications. For example, in reduced-order mod-

eling, a data-driven approach desires to map a full model trajectory to a reduced model trajectory or vice versa (Peherstorfer and Willcox, 2016). In solving parametric partial differential equations (PDEs), it is desired to learn a map from the parametric function space to the PDE solution space (Khoo et al., 2021; Li et al., 2020; Lu et al., 2021). In forward and inverse scattering problems (Khoo and Ying, 2019; Wei and Chen, 2019), it is interesting to learn an operator mapping the observed data function space to the parametric function space that models the underlying PDE. In density functional theory, it is desired to learn a nonlinear operator mapping a potential function to a density function (Fan et al., 2019a). In phase retrieval (Deng et al., 2020), an operator from the observed data function space to the reconstructed image function space is learned. Other image processing problems, e.g., image super-resolution (Qiao et al., 2021), image denoising (Tian et al., 2020), image inpainting (Qin et al., 2021), are similar to the deep learning-based phase retrieval, where an operator from a function space to another function space is learned.

As a powerful tool of nonparametric estimation, deep learning (Goodfellow et al., 2016) has made astonishing breakthroughs in various applications, including computer vision (Krizhevsky et al., 2012), natural language processing (Graves et al., 2013), speech recognition (Hinton et al., 2012), healthcare (Miotto et al., 2017), as well as nonlinear operator learning (Khoo et al., 2021; Zhu and Zabaras, 2018; Fan et al., 2019a,b; Khoo and Ying, 2019; Chen and Chen, 1995; Lu et al., 2021; Lanthaler et al., 2022; Bhattacharya et al., 2021; Li et al., 2020; Nelsen and Stuart, 2020; Kovachki et al., 2023; Zhang et al., 2023b,a). A typical method for operator learning is to first discretize the function spaces and represent each function by a vector using sampling. Then deep neural networks are applied to learn the map between these vector spaces (Khoo et al., 2021; Zhu and Zabaras, 2018; Fan et al., 2019a,b; Khoo and Ying, 2019). Such methods are mesh dependent: if a different discretization scheme is used, the network needs to be trained again. Though empirical successes have been demonstrated in learning nonlinear operators by this approach in many applications, it is computationally expensive to train these algorithms and the training procedure has to be repeated when the dimension of vector spaces is changed. Another approach based on the theory of approximating operators by neural networks (Chen and Chen, 1995) can alleviate this issue to a certain extent by avoiding the discretization of the output Hilbert space of the operator. This approach was first proposed in Chen and Chen (1995) with two-layer neural networks and recently revisited with deeper neural networks in Lu et al. (2021) with successful applications (Lin et al., 2021; Cai et al., 2021). However, the methods in Chen and Chen (1995); Lu et al. (2021); Lin et al. (2021); Cai et al. (2021) are still mesh-dependent due to the requirement of a fixed number of sample points for the input function of the operator. More recently, a discretization-invariant (mesh-independent) operator learning framework was proposed in Anandkumar et al. (2020); Bhattacharya et al. (2021); Li et al. (2020); Nelsen and Stuart (2020); Kovachki et al. (2023) by taking the advantage of graph kernel networks, principal component analysis (PCA), and kernel integral

operators, etc. With discretization-invariant approaches, the training procedure does not need to be performed again when the discretization scheme changes. The approximation ability of this neural operator learning framework is studied in Kovachki et al. (2023). For any given approximation error ε, the existence of a neural operator is proved to achieve the ε approximation error (Kovachki et al., 2023), while the scale of network size is not specified.

Although operator learning via deep learning-based nonparametric estimation has been successful in many applications, its statistical learning theory is still in its infancy, especially when the operator is from an infinite dimensional space to another. The successes of deep neural networks are largely due to their universal approximation power (Cybenko, 1989; Hornik, 1991), showing the existence of a neural network with a proper size fulfilling the approximation task for certain function classes. Quantitative function approximation theories, provably better than traditional tools, have been extensively studied with various network architectures and activation functions, e.g., for continuous functions (Yarotsky, 2017; Shen et al., 2020, 2021a,b, to appear; Yarotsky, 2021), for functions with certain smoothness (Yarotsky, 2018; Yarotsky and Zhevnerchuk, 2020; Lu et al., 2021; Suzuki, 2018), and for functions with integral representations (Barron, 1993; E et al., 2019, 2021; Siegel and Xu, 2021). In theory, deep neural networks can approximate certain high dimensional functions with a fast rate that is independent of the input dimension (Barron, 1993; E et al., 2019, 2021; Siegel and Xu, 2021; Shen et al., 2021a,b; Yarotsky and Zhevnerchuk, 2020; Shen et al., 2021c; Chen and Chen, 1995; Chen et al., 2022, 2020; Liu et al., 2021; Jiao et al., 2021; Cloninger and Klock, 2020; Shaham et al., 2018; Schmidt-Hieber, 2019; Du et al., 2021; Nakada and Imaizumi, 2020b; Liu et al., 2022, 2024). However, in the context of operator approximation, deep learning theory is very limited. Probably the first result is the universal approximation theorem for operators in Chen and Chen (1995). More recently, theories on local approximation of operators by deep neural networks were studied in Mhaskar (2022). Quantitative approximation results for operators between infinite dimensional spaces were given in Bhattacharya et al. (2021); Lanthaler et al. (2022); Kovachki et al. (2021) based on the function approximation theory in Yarotsky (2017). Note that the function approximation results in Yarotsky (2017) does not give the flexibility to choose arbitrary width and depth of neural networks. In this paper, we provide a new operator approximation theory based on nearly optimal function approximation results where the width and depth of the network can be chosen flexibly. In comparison with Lanthaler et al. (2022), the flexibility of choosing arbitrary width and depth provides an explicit guideline to balance the approximation error and the statistical variance to achieve a better generalization error in operator learning.

We also establish a novel statistical theory for deep nonparametric estimation of Lipschitz operators between infinite dimensional Hilbert spaces. The core question to be answered is: how the generalization error scales when the number of training samples increases

and whether the scaling is dimension-independent without the curse of dimensionality. In literature, the statistical theory for function regression via neural networks has been a popular research topic (Hamers and Kohler, 2006; Kohler and Krzyżak, 2005; Jacot et al., 2018; Bauer and Kohler, 2019; Schmidt-Hieber, 2020; Cao and Gu, 2019; Chen et al., 2022; Kohler et al., 2020; Nakada and Imaizumi, 2020a; Farrell et al., 2021; Liu et al., 2021; Jiao et al., 2021). These works have proved that deep nonparametric regression can achieve the optimal minimax rate of regression established in Stone (1982); Györfi et al. (2002). When the target function has low complexity or the function domain is a low dimensional set, deep neural networks can achieve a fast rate depending on the intrinsic dimension (Chen et al., 2019, 2022, 2020; Liu et al., 2021; Shen et al., 2020; Jiao et al., 2021; Cloninger and Klock, 2020; Shaham et al., 2018; Schmidt-Hieber, 2019; Du et al., 2021; Nakada and Imaizumi, 2020b). In more sophisticated cases when a mathematical modeling problem is transferred to a special regression problem, e.g., solving high dimensional PDEs and identifying the governing equation of spatial-temporal data, the generalization analysis of deep learning has been proposed in Berner et al. (2018); Shin et al. (2020); Luo and Yang (2020); Mishra and Molinaro (2020); Lu et al. (2021); Lu and Lu (2021); Duan et al. (2021); Gu et al. (2021). All these results focus on the regression problem when the target function is a mapping from a finite dimensional space to a finite dimensional space. Therefore, these results cannot be applied to mappings from an infinite dimensional space to another. To our best knowledge, the only work on the generalization error analysis of deep operator learning in Hilbert spaces is Lanthaler et al. (2022) for the algorithm in Lu et al. (2021), which is not completely discretization-invariant. The generalization error in Lanthaler et al. (2022) is a posteriori depending on the properties of neural networks fitting the target operator. Recently, the posterior rates on learning linear operators by Bayesian inversion have been studied in de Hoop et al. (2021).

In this paper, we establish a priori generalization error for a discretization-invariant operator learning algorithm for operators between Hilbert spaces. As we shall see later, our theory can be applied to operator learning from a finite dimensional vector space to another as a special case. Therefore, the theoretical result in this paper can facilitate the understanding of many operator learning algorithms by neural networks in the literature. Our contributions are summarized as follows:

1. We derive an upper bound on the generalization error for a general framework of learning operators between infinite dimensional spaces by deep neural networks. The framework considered here first encodes the input and output space into finite-dimensional spaces by some encoders and decoders. Then a transformation between the dimension reduced spaces is learned using deep neural networks. Our upper bound is derived for two network architectures: one has constraints on the number of nonzero weight parameters and parameter magnitude; The other network architecture does not have such constraints and allows one to flexibly choose the depth and width. Our upper

bound consists of two parts: the error of learning the transformation by deep neural networks, and the dimension reduction error with encoders and decoders.

2. Our analysis is general and can be applied for a wide range of popular choices of encoders and decoders in the numerical implementation, such as those derived from Legendre polynomials, trigonometric bases, and principal component analysis. The generalization error is given for each of these examples.

3. We discuss two scenarios to further exploit the additional low-dimensional structures of data in operator estimation motivated by practical considerations and classical numerical methods. The first scenario is when encoded vectors in the input space are on a low-dimensional manifold. In this scenario, we show that the generalization error converges as the training sample increases with a fast rate depending on the intrinsic dimension of the manifold. The second scenario is when the operator itself has low complexity: the composition of the operator with a certain encoder and decoder is a multi-index model. In this scenario, we show that the convergence rate of the generalization error depends on the intrinsic dimension of the composed operator.

We organize this paper as follows. In Section 2, we introduce our notations and the learning framework considered in this paper. Our main results with general encoders and decoders are presented in Section 3. We discuss the applications of our main results to specific encoders and decoders derived from certain function basis and PCA in Section 4 and 5, respectively. To further exploit additional low-dimensional structures of data, we discuss the application of our results to two scenarios in Section 6. The proofs of all results are given in Section 7. We conclude this paper in Section 8.

2. A general framework

In this section, we introduce the framework considered in this paper for learning operators between infinite dimensional spaces.

2.1 Preliminaries

We first briefly introduce some definitions and notations on a Hilbert space, encoders, decoders, and feedforward neural networks used in this paper. A Hilbert space is a Banach space equipped with an inner product. It is separable if it admits a countable orthonormal basis. Let $\mathcal{H}$ be a separable Hilbert space. An encoder for $\mathcal{H}$ is an operator $E_{\mathcal{H}} : \mathcal{H} \to \mathbb{R}^d$, where d is a positive integer representing the encoding dimension. The associated decoder is an operator $D_{\mathcal{H}} : \mathbb{R}^d \to \mathcal{H}$. The composition $\Pi_{\mathcal{H}} = D_{\mathcal{H}} \circ E_{\mathcal{H}} : \mathcal{H} \to \mathcal{H}$ is a projection. For any $u \in \mathcal{H}$, we define the projection error as $\|\Pi_{\mathcal{H}}(u) - u\|_{\mathcal{H}}$.

In this paper, we consider the ReLU Feedforward Neural Network (FNN) in the form of

$$f(\mathbf{x}) = W_L \cdot \mathrm{ReLU}\left(W_{L-1} \cdots \mathrm{ReLU}(W_1\mathbf{x} + \mathbf{b}_1) + \cdots + \mathbf{b}_{L-1}\right) + \mathbf{b}_L, \tag{1}$$

where W_l's are weight matrices, $\mathbf{b}_l$'s are biases, and $\mathrm{ReLU}(a) = \max\{a, 0\}$ is the rectified linear unit activation (ReLU) applied element-wise.

We consider two classes of network architectures whose inputs are in a compact domain of a vector space and whose outputs are vectors in $\mathbb{R}^d$. The dimension of the input and output spaces are to be specified later. The first class is defined as

$$\begin{aligned}
&\mathcal{F}_{\mathrm{NN}}(d, L, p, K, \kappa, M) \\
&= \{\Gamma = [f_1, f_2, ..., f_d]^\top : \text{ for each } k = 1, ..., d, \\
&\quad f_k(\mathbf{x}) \text{ is in the form of (1) with } L \text{ layers, width bounded by } p, \\
&\quad \|f_k\|_\infty \le M,\ \|W_l\|_{\infty,\infty} \le \kappa,\ \|\mathbf{b}_l\|_\infty \le \kappa,\ \sum_{l=1}^{L} \|W_l\|_0 + \|\mathbf{b}_l\|_0 \le K\},
\end{aligned} \tag{2}$$

where $\|f\|_\infty = \sup_{\mathbf{x}} |f(\mathbf{x})|$, $\|W\|_{\infty,\infty} = \max_{i,j} |W_{i,j}|$, $\|\mathbf{b}\|_\infty = \max_i |b_i|$ for any function f, matrix W, and vector $\mathbf{b}$ with $\|\cdot\|_0$ denoting the number of nonzero elements of its argument. The function class given by this first network architecture has an upper bound on all weight parameters (the magnitude of all weight parameters are upper bounded by κ) and a cardinality constraint (the total number of nonzero parameters are no more than K). Each element of the output is upper bounded by M. This constraint on the output is often enforced by clipping the output in the testing procedure. Such a clipping can be realized with a two-layer network, which is fixed during training. This clipping step is common in nonparametric regression (Györfi et al., 2002).

In the second class of network architecture, we drop the magnitude and cardinality constraints for practical concerns on training. The second network architecture is parameterized by L, p, M only:

$$\begin{aligned}
\mathcal{F}_{\mathrm{NN}}(d, L, p, M) = \{\Gamma =& [f_1, f_2, ..., f_d]^\top : \text{ for each } k = 1, ..., d, \\
& f_k(\mathbf{x}) \text{ is in the form of (1) with } L \text{ layers, width bounded by } p, \\
& \|f_k\|_\infty \le M\}.
\end{aligned} \tag{3}$$

All theoretical results in this paper can be applied to both network architectures.

Notations: We use bold lowercase letters to denote vectors, and normal font letters to denote scalars. The notation $\mathbf{0}$ represents a zero vector. For a d dimensional vector $\mathbf{k} = [k_1, \cdots, k_d]^\top$, we denote $|\mathbf{k}| = \sum_{i=1}^d k_i$. The vector norms are defined as $\|\mathbf{k}\|_\infty = \max_i |k_i|$ and $\|\mathbf{k}\|_2 = \sqrt{\sum_{i=1}^d k_i^2}$. For any scalar s, we denote $\lceil s \rceil$ as the smallest integer that is no less than s. We use $\mathbb{N}$ to denote the set of positive integers and $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$. For a function $f : \Omega \to \mathbb{R}$ in a Hilbert space $\mathcal{H}$, we define the function norms as $\|f\|_\infty = \sup_{\mathbf{x}\in\Omega} |f(\mathbf{x})|$ and $\|f\|_{\mathcal{H}} = \sqrt{\langle f, f\rangle_{\mathcal{H}}}$, where $\langle\cdot,\cdot\rangle_{\mathcal{H}}$ denotes the inner product of $\mathcal{H}$. For an operator $A : \mathcal{H} \to \mathcal{H}$, we denote its operator norm by $\|A\|_{\mathrm{op}}$ and its Hilbert-Schmidt norm by $\|A\|_{\mathrm{HS}}$. More notations used in this paper is summarized in Table 1.

2.2 Problem setup and a learning framework

Let $\mathcal{X}$ and $\mathcal{Y}$ be two separable Hilbert spaces and $\Psi : \mathcal{X} \to \mathcal{Y}$ be an unknown operator. Our goal is to learn the operator Ψ from a finite number of samples $\mathcal{S} = \{u_i, v_i\}_{i=1}^{2n}$ in the following setting.

Setting 1 *Let $\mathcal{X}, \mathcal{Y}$ be two separable Hilbert spaces and γ be a probability measure on $\mathcal{X}$. Let $\mathcal{S} = \{u_i, v_i\}_{i=1}^{2n}$ be the given data where u_i's are i.i.d. samples from γ and the v_i's are generated according to model:*

$$v_i = \Psi(u_i) + \widetilde{\epsilon}_i, \tag{4}$$

where the $\widetilde{\epsilon}_i$'s are i.i.d. samples from a probability measure μ on $\mathcal{Y}$, independently of u_i's. We denote the probability measure of v by ζ.

The pushforward measure of γ under Ψ is denoted by $\Psi_{\#}\gamma$, such that for any $\Omega \subset \mathcal{Y}$,

$$\Psi_{\#}\gamma(\Omega) = \gamma\left(\{u : \Psi(u) \in \Omega\}\right).$$

Without additional assumptions, the estimation error of Ψ based on a finite number of samples may not converge to zero since Ψ is an operator between infinite-dimensional spaces. In this paper, we exploit the low-dimensional structures in this estimation problem arising from practical applications, and prove a nonparametric estimation error of Ψ by deep neural networks.

Our learning framework follows the idea of model reduction in Bhattacharya et al. (2021). It consists of encoding and decoding in both the $\mathcal{X}$ and $\mathcal{Y}$ spaces, and deep learning of a transformation between the encoded vectors for the elements in $\mathcal{X}$ and $\mathcal{Y}$. We first encode the elements in $\mathcal{X}$ and $\mathcal{Y}$ to finite dimensional vectors by an encoding operator. For fixed positive integers $d_{\mathcal{X}}$ and $d_{\mathcal{Y}}$, let $E_{\mathcal{X}} : \mathcal{X} \to \mathbb{R}^{d_{\mathcal{X}}}$ and $D_{\mathcal{X}} : \mathbb{R}^{d_{\mathcal{X}}} \to \mathcal{X}$ be the encoder and decoder of $\mathcal{X}$, and $E_{\mathcal{Y}} : \mathcal{Y} \to \mathbb{R}^{d_{\mathcal{Y}}}$ and $D_{\mathcal{Y}} : \mathbb{R}^{d_{\mathcal{Y}}} \to \mathcal{Y}$ be the encoder and decoder of $\mathcal{Y}$ such that

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

The empirical counterparts of encoders and decoders are denoted by $E_{\mathcal{X}}^n, D_{\mathcal{X}}^n, E_{\mathcal{Y}}^n$ and $D_{\mathcal{Y}}^n$, and we call them empirical encoders and decoders.

The simplest encoder in a function space is the discretization operator. When $\mathcal{X}$ is a function space containing functions defined on a compact subset of $\mathbb{R}^D$, we can discretize the domain with a fixed grid, and take the encoder as the sampling operator on this grid. However, the discretization operator may not reveal the low-dimensional structures in the functions of interest, and therefore may not effectively reduce the dimension.

A popular choice of encoders in applications is the basis encoder, such as the Fourier transform with trigonometric basis, or PCA with data-driven basis, etc. Given an orthonormal basis of $\mathcal{X}$ and a positive integer $d_{\mathcal{X}}$, the basis encoder maps an element in $\mathcal{X}$ to $d_{\mathcal{X}}$

coefficients associated with a fixed set of $d_{\mathcal{X}}$ bases. For any coefficient vector $\mathbf{a} \in \mathbb{R}^{d_{\mathcal{X}}}$, the decoder $D_{\mathcal{X}}(\mathbf{a})$ gives rise to a linear combination of these $d_{\mathcal{X}}$ bases weighted by $\mathbf{a}$. See Section 4 for the details about the basis encoder. The trigonometric basis and orthogonal polynomials are commonly used bases in applications. These bases are a priori given, independently of the training data. In this case, the basis encoder can be viewed as a deterministic operator, which is given independently of the training data. The empirical encoder and decoder are the same as the oracle encoder and decoder: $E_{\mathcal{X}}^n = E_{\mathcal{X}}$ and $D_{\mathcal{X}}^n = D_{\mathcal{X}}$.

PCA (Pearson, 1901; Hotelling, 1933, 1992) is an effective dimension reduction technique, when u_i's exhibit a low-dimensional linear structure. The PCA encoder encodes an element in $\mathcal{X}$ to the $d_{\mathcal{X}}$ coefficients associated with the top $d_{\mathcal{X}}$ eigenbasis of a trace operator. The PCA decoder gives a linear combination of the eigenbasis weighted by the given coefficient vector. In practice, one needs to estimate this trace operator from the training data and obtain an empirical estimation of $E_{\mathcal{X}}$ and $D_{\mathcal{X}}$, which are denoted by $E_{\mathcal{X}}^n$ and $D_{\mathcal{X}}^n$, respectively. The PCA encoder is data-driven, and we expect $E_{\mathcal{X}}^n \approx E_{\mathcal{X}},\ D_{\mathcal{X}}^n \approx D_{\mathcal{X}}$ when the sample size n is sufficiently large. The encoding and decoding operator in $\mathcal{Y}$ can be defined analogously.

The operator $D_{\mathcal{X}} \circ E_{\mathcal{X}}$ is the projection operator associated with the encoder $E_{\mathcal{X}}$ and decoder $D_{\mathcal{X}}$. We have the following projections and their empirical counterparts:

$$\Pi_{\mathcal{X},d_{\mathcal{X}}} = D_{\mathcal{X}} \circ E_{\mathcal{X}}, \quad \Pi_{\mathcal{X},d_{\mathcal{X}}}^n = D_{\mathcal{X}}^n \circ E_{\mathcal{X}}^n,$$
$$\Pi_{\mathcal{Y},d_{\mathcal{Y}}} = D_{\mathcal{Y}} \circ E_{\mathcal{Y}}, \quad \Pi_{\mathcal{Y},d_{\mathcal{Y}}}^n = D_{\mathcal{Y}}^n \circ E_{\mathcal{Y}}^n.$$

After the empirical encoders $E_{\mathcal{X}}^n, E_{\mathcal{Y}}^n$ and decoders $D_{\mathcal{X}}^n, D_{\mathcal{Y}}^n$ are computed, our objective is to learn a transformation $\Gamma : \mathbb{R}^{d_{\mathcal{X}}} \to \mathbb{R}^{d_{\mathcal{Y}}}$ such that

$$D_{\mathcal{Y}}^n \circ \Gamma \circ E_{\mathcal{X}}^n \approx \Psi. \tag{5}$$

We learn Γ using a two-stage algorithm. Given the training data $\mathcal{S} = \{u_i, v_i\}_{i=1}^{2n}$, we split the data into two subsets $\mathcal{S}_1 = \{u_i, v_i\}_{i=1}^{n}$ and $\mathcal{S}_2 = \{u_i, v_i\}_{i=n+1}^{2n}$ [1], where $\mathcal{S}_1$ is used to compute the encoders and decoders and $\mathcal{S}_2$ is used to learn the transformation Γ between the encoded vectors. Our two-stage algorithm follows

Stage 1: Compute the empirical encoders and decoders $E_{\mathcal{X}}^n, D_{\mathcal{X}}^n, E_{\mathcal{Y}}^n, D_{\mathcal{Y}}^n$ based on $\mathcal{S}_1$. In the case of deterministic encoders, we skip Stage 1 and let $E_{\mathcal{X}}^n = E_{\mathcal{X}},\ D_{\mathcal{X}}^n = D_{\mathcal{X}},\ E_{\mathcal{Y}}^n = E_{\mathcal{Y}},\ D_{\mathcal{Y}}^n = D_{\mathcal{Y}}$.

Stage 2: Learn Γ with $\mathcal{S}_2$ by solving the following optimization problem

$$\Gamma_{\mathrm{NN}} \in \operatorname*{argmin}_{\Gamma \in \mathcal{F}_{\mathrm{NN}}} \frac{1}{n} \sum_{i=n+1}^{2n} \|\Gamma \circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n(v_i)\|_2^2 \tag{6}$$

for some $\mathcal{F}_{\mathrm{NN}}$ class with a proper choice of parameters.

1. The data can be split unevenly as well.

Notation	Description	Notation	Description
$\mathcal{X}$	Input space	$\mathcal{Y}$	Output space
$\Psi : \mathcal{X} \to \mathcal{Y}$	An unknown operator	$\mathcal{S} = \{u_i, v_i\}_{i=1}^{2n}$	Given data set
γ	A probability measure on $\mathcal{X}$	$\Psi_{\#}\gamma$	Push forward measure of γ under Ψ
μ	The probability measure of noise $\widetilde{\varepsilon}$	ζ	The probability measure of $v = \Psi(u) + \widetilde{\varepsilon}$
$E_{\mathcal{X}}, D_{\mathcal{X}}$	Encoder and decoder of $\mathcal{X}$	$E_{\mathcal{Y}}, D_{\mathcal{Y}}$	Encoder and decoder of $\mathcal{Y}$
$E_{\mathcal{X}}^n, D_{\mathcal{X}}^n$	Empirical estimations of $E_{\mathcal{X}}, D_{\mathcal{X}}$ from noisy data	$E_{\mathcal{Y}}^n, D_{\mathcal{Y}}^n$	Empirical estimations of $E_{\mathcal{Y}}, D_{\mathcal{Y}}$ from noisy data
$d_{\mathcal{X}}$	Encoding dimension of $\mathcal{X}$	$d_{\mathcal{Y}}$	Encoding dimension of $\mathcal{Y}$
$\Pi_{\mathcal{X},d_{\mathcal{X}}}$	Projection $D_{\mathcal{X}} \circ E_{\mathcal{X}}$	$\Pi_{\mathcal{Y},d_{\mathcal{Y}}}$	Projection $D_{\mathcal{Y}} \circ E_{\mathcal{Y}}$
$\Pi_{\mathcal{X},d_{\mathcal{X}}}^n$	Empirical projection $D_{\mathcal{X}}^n \circ E_{\mathcal{X}}^n$	$\Pi_{\mathcal{Y},d_{\mathcal{Y}}}^n$	Empirical projection $D_{\mathcal{Y}}^n \circ E_{\mathcal{Y}}^n$
$\|\Pi_{\mathcal{X},d_{\mathcal{X}}}(u) - u\|_{\mathcal{X}}$	Encoding error for u in $\mathcal{X}$	$\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}(v) - v\|_{\mathcal{Y}}$	Encoding error for v in $\mathcal{Y}$
$\mathcal{F}_{\text{NN}}$	Neural network class	Γ_{NN}	Neural network estimator in (6)

Table 1: Notations used in this paper.

Our estimator of Ψ, a neural operator, is given as

$$\Psi_{\text{NN}} := D_{\mathcal{Y}}^n \circ \Gamma_{\text{NN}} \circ E_{\mathcal{X}}^n,$$

and the mean squared generalization error is defined as

$$\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|\Psi_{\text{NN}}(u) - \Psi(u)\|_{\mathcal{Y}}^2. \tag{7}$$

In (6), the transformation Γ_{NN} is learned by minimizing the mean squared error in the encoded space $\mathbb{R}^{d_{\mathcal{Y}}}$. In literature (Bhattacharya et al., 2021), another loss that are popularly adopted is the following one

$$\Gamma_{\text{NN}} \in \underset{\Gamma\in\mathcal{F}_{\text{NN}}}{\operatorname{argmin}} \frac{1}{n}\sum_{i=n+1}^{2n} \|D_{\mathcal{Y}}^n \circ \Gamma \circ E_{\mathcal{X}}^n(u_i) - v_i\|_{\mathcal{Y}}^2, \tag{8}$$

in which the transformation Γ_{NN} is learned by minimizing the mean squared error in the output space $\mathcal{Y}$. In this paper, we will derive upper bounds of the generalization error with the loss defined in (6). An upper bound of the generalization error with the loss (8) can be derived with similar techniques.

3. Main results

The main results of this paper provide statistical guarantees on the mean squared generalization error for the estimation of Lipchitz operators.

3.1 Assumptions

We first make some assumptions on the measure γ and the operator Ψ.

Assumption 1 (Compactly supported measure) *The probability distribution γ is supported on a compact set $\Omega_{\mathcal{X}} \subset \mathcal{X}$. There exists $R_{\mathcal{X}} > 0$ such that, for any $u \in \Omega_{\mathcal{X}}$, we have*

$$\|u\|_{\mathcal{X}} \leq R_{\mathcal{X}}.$$

Assumption 2 (Lipschitz operator) *There exists $L_\Psi > 0$ such that*

$$\|\Psi(u_1) - \Psi(u_2)\|_{\mathcal{Y}} \leq L_\Psi \|u_1 - u_2\|_{\mathcal{X}}, \quad \text{for any } u_1, u_2 \in \Omega_{\mathcal{X}}.$$

Assumption 1 and 2 assume that γ is compactly supported and Ψ is Lipschitz continuous. We denote the image of $\Omega_{\mathcal{X}}$ under the transformation Ψ as

$$\Omega_{\mathcal{Y}} = \{v \in \mathcal{Y} : v = \Psi(u) \text{ for some } u \in \Omega_{\mathcal{X}}\}.$$

Assumption 1 and 2 imply that $\Omega_{\mathcal{Y}}$ is bounded: there exists a constant $R_{\mathcal{Y}} > 0$ depending on $R_{\mathcal{X}}$ and L_Ψ such that for any $v \in \Omega_{\mathcal{Y}}$, we have $\|v\|_{\mathcal{Y}} \leq R_{\mathcal{Y}}$.

We next make some natural assumptions on the empirical encoders and decoders:

Assumption 3 (Lipchitz encoders and decoders) *The empirical encoders and decoders $E^n_{\mathcal{X}}, D^n_{\mathcal{X}}, E^n_{\mathcal{Y}}, D^n_{\mathcal{Y}}$ satisfy:*

$$E^n_{\mathcal{X}}(0_{\mathcal{X}}) = \mathbf{0},\ D^n_{\mathcal{X}}(\mathbf{0}) = 0_{\mathcal{X}},\ E^n_{\mathcal{Y}}(0_{\mathcal{Y}}) = \mathbf{0},\ D^n_{\mathcal{Y}}(\mathbf{0}) = 0_{\mathcal{Y}},$$

where $\mathbf{0}$ denotes the zero vector, $0_{\mathcal{X}}$ is the zero function in $\mathcal{X}$ and $0_{\mathcal{Y}}$ is the zero function in $\mathcal{Y}$.

They are also Lipschitz: there exist $L_{E^n_{\mathcal{X}}}, L_{D^n_{\mathcal{X}}}, L_{E^n_{\mathcal{Y}}}, L_{D^n_{\mathcal{Y}}} > 0$ such that, for any $u_1, u_2 \in \mathcal{X}$ and any $\mathbf{a}_1, \mathbf{a}_2 \in \mathbb{R}^{d_{\mathcal{X}}}$, we have

$$\|E^n_{\mathcal{X}}(u_1) - E^n_{\mathcal{X}}(u_2)\|_2 \leq L_{E^n_{\mathcal{X}}} \|u_1 - u_2\|_{\mathcal{X}}, \quad \|D^n_{\mathcal{X}}(\mathbf{a}_1) - D^n_{\mathcal{X}}(\mathbf{a}_2)\|_{\mathcal{X}} \leq L_{D^n_{\mathcal{X}}} \|\mathbf{a}_1 - \mathbf{a}_2\|_2,$$

and for any $v_1, v_2 \in \mathcal{Y}$ and any $\mathbf{a}_1, \mathbf{a}_2 \in \mathbb{R}^{d_{\mathcal{Y}}}$, we have

$$\|E^n_{\mathcal{Y}}(v_1) - E^n_{\mathcal{Y}}(v_2)\|_2 \leq L_{E^n_{\mathcal{Y}}} \|v_1 - v_2\|_{\mathcal{Y}}, \quad \|D^n_{\mathcal{Y}}(\mathbf{a}_1) - D^n_{\mathcal{Y}}(\mathbf{a}_2)\|_{\mathcal{Y}} \leq L_{D^n_{\mathcal{Y}}} \|\mathbf{a}_1 - \mathbf{a}_2\|_2.$$

Remark 1 *Assumption 3 is made on empirical encoders and decoders. The basis encoders and PCA encoders, which are most commonly used, satisfy Assumption 3 with the Lipchitz constants $L_{E^n_{\mathcal{X}}} = L_{D^n_{\mathcal{X}}} = L_{E^n_{\mathcal{Y}}} = L_{D^n_{\mathcal{Y}}} = 1$, independently of the training data (see Lemma 6 and Lemma 12).*

Assumption 3 implies that $E^n_{\mathcal{X}}(u)$ and $E^n_{\mathcal{Y}}(v)$ are bounded for any $u \in \Omega_{\mathcal{X}}$ and $v \in \Omega_{\mathcal{Y}}$. For any $u \in \Omega_{\mathcal{X}}$, we have $\|E^n_{\mathcal{X}}(u)\|_2 \leq \|E^n_{\mathcal{X}}(u) - E^n_{\mathcal{X}}(0)\|_2 + \|E^n_{\mathcal{X}}(0)\|_2 \leq L_{E^n_{\mathcal{X}}} R_{\mathcal{X}}$. Similarly, for any $v \in \Omega_{\mathcal{Y}}$, we have $\|E^n_{\mathcal{Y}}(v)\|_2 \leq L_{E^n_{\mathcal{Y}}} R_{\mathcal{Y}}$.

Remark 2 *The condition*

$$E^n_{\mathcal{X}}(0_{\mathcal{X}}) = \mathbf{0},\ D^n_{\mathcal{X}}(\mathbf{0}) = 0_{\mathcal{X}},\ E^n_{\mathcal{Y}}(0_{\mathcal{Y}}) = \mathbf{0},\ D^n_{\mathcal{Y}}(\mathbf{0}) = 0_{\mathcal{Y}},$$

in Assumption 3 is only used to make sure $E^n_{\mathcal{X}}(u)$ and $E^n_{\mathcal{Y}}(v)$ are bounded. One can replace $\mathbf{0}$ *by any finite vector.*

Assumption 4 (Noise) *The random noise $\widetilde{\epsilon}$ satisfies*

(i) $\widetilde{\epsilon}$ is independent of u.

(ii) $\mathbb{E}[\widetilde{\epsilon}] = 0$.

(iii) There exists $\widetilde{\sigma} > 0$ such that $\|\widetilde{\epsilon}\|_{\mathcal{Y}} \leq \widetilde{\sigma}$.

Assumption 4(i)-(iii) are natural assumptions on noise. Assumption 4(i) is about the independence of the input and the noise, which is commonly used in nonparametric regression. Assumption 4(iii) together with Assumption 3 imply that the perturbation of the encoded vectors are bounded: $\|E^n_{\mathcal{Y}}(\Psi(u)+\widetilde{\epsilon}) - E^n_{\mathcal{Y}}(\Psi(u))\|_\infty \leq L_{E^n_{\mathcal{Y}}}\widetilde{\sigma}$. We denote $\sigma = L_{E^n_{\mathcal{Y}}}\widetilde{\sigma}$ such that

$$\|E^n_{\mathcal{Y}}(\Psi(u) + \widetilde{\epsilon}) - E^n_{\mathcal{Y}}(\Psi(u))\|_\infty \leq \sigma \text{ for any } u \text{ and } \widetilde{\epsilon}. \tag{9}$$

Assumption 5 (Noise and encoder) *For any noise satisfying Assumption 4 and any given $\mathcal{S}_1$, the conditional expectation satisfies*

$$\mathbb{E}_{\widetilde{\epsilon}}\left[E^n_{\mathcal{Y}}(\Psi(u) + \widetilde{\epsilon}) - E^n_{\mathcal{Y}}(\Psi(u))|\mathcal{S}_1\right] = \mathbf{0},\ \textit{for any } u \in \Omega_{\mathcal{X}},$$

where $E^n_{\mathcal{Y}}$ is the empirical encoder computed with $\mathcal{S}_1$.

Assumption 5 requires that, if we condition on $\mathcal{S}_1$ based on which the empirical encoder $E^n_{\mathcal{Y}}$ is computed, the perturbation on the encoded vector resulted from noise has zero expectation. Assumption 5 is guaranteed for all linear encoders as long as Assumption 4(ii) holds:

$$\mathbb{E}_{\widetilde{\epsilon}}\left[E^n_{\mathcal{Y}}(\Psi(u) + \widetilde{\epsilon}) - E^n_{\mathcal{Y}}(\Psi(u))|\mathcal{S}_1\right] = \mathbb{E}_{\widetilde{\epsilon}}\left[E^n_{\mathcal{Y}}(\widetilde{\epsilon})|\mathcal{S}_1\right] = \mathbf{0}.$$

Basis encoders, including the PCA encoder, are linear encoders, so they all satisfy Assumption 5.

3.2 Generalization error with general encoders and decoders

Our main result is an upper bound of the generalization error in (7) with general encoders and decoders. Our results can be applied to both network architectures defined in (2) and (3). Our first theorem gives an upper bound of the generalization error with the network architecture defined in (2).

Theorem 3 *In Setting 1, suppose Assumption 1 – 5 hold. Let* Γ_{NN} *be the minimizer of (6) with the network architecture* $\mathcal{F}(d_{\mathcal{Y}}, L, p, K, \kappa, M)$ *in (2), where*

$$\begin{aligned} &L = O(\log n + \log d_{\mathcal{Y}}),\ p = O\left(d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{2+d_{\mathcal{X}}}}\right),\ K = O\left(d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} \log n\right), \\ &M = \sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}},\ \kappa = \max\left\{1, \sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}, \sqrt{d_{\mathcal{X}}} L_{E_{\mathcal{X}}^n} R_{\mathcal{X}}, L_{E_{\mathcal{Y}}^n} L_{D_{\mathcal{X}}^n} L_{\Psi}\right\}. \end{aligned} \tag{10}$$

Then we have

$$\begin{aligned} &\mathbb{E}_{\mathcal{S}} \mathbb{E}_{u\sim\gamma} \| D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u) \|_{\mathcal{Y}}^2 \\ \leq\ & C_1 (\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^3 n + C_2 (\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^2 (\log d_{\mathcal{Y}}) n^{-1} \\ &+ C_3 \mathbb{E}_{\mathcal{S}} \mathbb{E}_{u\sim\gamma} \| \Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u) - u \|_{\mathcal{X}}^2 + 2 \mathbb{E}_{\mathcal{S}} \mathbb{E}_{w\sim\Psi_{\#}\gamma} \| \Pi_{\mathcal{Y}, d_{\mathcal{Y}}}^n(w) - w \|_{\mathcal{Y}}^2, \end{aligned} \tag{11}$$

where C_1, C_2 *are constants depending on* $d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{E_{\mathcal{X}}^n}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}^n}, L_{D_{\mathcal{Y}}^n}, L_{\Psi}$ *and* $C_3 = 16 L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_{\Psi}^2$.

Our second theorem gives an upper bound of the generalization error with the network architecture defined in (3).

Theorem 4 *In Setting 1, suppose Assumption 1 – 5 hold. Let* Γ_{NN} *be the minimizer of (6) with the network architecture* $\mathcal{F}(d_{\mathcal{Y}}, L, p, M)$ *in (3) with*

$$L = O(\widetilde{L}),\ p = O\left(\widetilde{p}\right), M = \sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}, \tag{12}$$

where $\widetilde{L}, \widetilde{p} > 0$ *are positive integers satisfying*

$$\widetilde{L}\widetilde{p} = \left\lceil d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{4+2d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{4+2d_{\mathcal{X}}}} \right\rceil. \tag{13}$$

Then we have

$$\begin{aligned} &\mathbb{E}_{\mathcal{S}} \mathbb{E}_{u\sim\gamma} \| D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u) \|_{\mathcal{Y}}^2 \\ \leq\ & C_4 (\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^2 n + C_3 \mathbb{E}_{\mathcal{S}} \mathbb{E}_{u\sim\gamma} \| \Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u) - u \|_{\mathcal{X}}^2 \\ &+ 2 \mathbb{E}_{\mathcal{S}} \mathbb{E}_{w\sim\Psi_{\#}\gamma} \| \Pi_{\mathcal{Y}, d_{\mathcal{Y}}}^n(w) - w \|_{\mathcal{Y}}^2, \end{aligned} \tag{14}$$

where C_4 *is a constant depending on* $d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{E_{\mathcal{X}}^n}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}^n}, L_{\Psi}$ *and* $C_3 = 16 L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_{\Psi}^2$ *is the same one in Theorem 3.*

Remark 5 *In Assumption 2, the operator Ψ is assumed to be Lipscthiz. We can relax this assumption to Hölder continuous operators. Specifically, for $0 < \alpha \leq 1$, we assume that there exists $L_{\Psi,\alpha} > 0$ such that*

$$\|\Psi(u_1) - \Psi(u_2)\|_{\mathcal{Y}} \leq L_{\Psi,\alpha}\|u_1 - u_2\|_{\mathcal{X}}^{\alpha}, \quad \text{for any } u_1, u_2 \in \Omega_{\mathcal{X}}.$$

Under the Hölder assumption, we can prove a similar result with the same technique. Specifically, for Theorem 3, we can show that if we set

$$L = O(\log n + \log d_{\mathcal{Y}}),\ p = O\left(d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{2\alpha+d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{2\alpha+d_{\mathcal{X}}}}\right),\ K = O\left(d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{2\alpha+d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{2\alpha+d_{\mathcal{X}}}} \log n\right),$$
$$M = \sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}},\ \kappa = \max\left\{1, \sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}, \sqrt{d_{\mathcal{X}}} L_{E_{\mathcal{X}}^n} R_{\mathcal{X}}, L_{E_{\mathcal{Y}}^n} L_{D_{\mathcal{X}}^n} L_{\Psi}\right\}.$$

Then we have

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u)\|_{\mathcal{Y}}^2 \\
\leq\ & C_1'(\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4\alpha+d_{\mathcal{X}}}{2\alpha+d_{\mathcal{X}}}} n^{-\frac{2\alpha}{2\alpha+d_{\mathcal{X}}}} \log^3 n + C_2'(\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^2 (\log d_{\mathcal{Y}}) n^{-1} \\
&+ C_3'\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|\Pi_{\mathcal{X},d_{\mathcal{X}}}^n(u) - u\|_{\mathcal{X}}^{2\alpha} + 2\mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}^n(w) - w\|_{\mathcal{Y}}^2,
\end{aligned}$$

where C_1', C_2' are constants depending on $\alpha, d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{E_{\mathcal{X}}^n}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}^n}, L_{D_{\mathcal{Y}}^n}, L_{\Psi}$ and $C_3' = 16L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_{\Psi}^2$.

For Theorem 4, we can show that if we choose $\widetilde{L}, \widetilde{p} > 0$ satisfying

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

then

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u)\|_{\mathcal{Y}}^2 \\
\leq\ & C_4'(\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4\alpha+d_{\mathcal{X}}}{2\alpha+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^2 n + C_3'\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|\Pi_{\mathcal{X},d_{\mathcal{X}}}^n(u) - u\|_{\mathcal{X}}^{2\alpha} \\
&+ 2\mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}^n(w) - w\|_{\mathcal{Y}}^2,
\end{aligned}$$

where C_4' is a constant depending on $\alpha, d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{E_{\mathcal{X}}^n}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}^n}, L_{\Psi}$ and $C_3' = 16L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_{\Psi}^2$.

Theorem 3 is proved in Section 7.2 and Theorem 4 is proved in Section 7.3. Theorem 3 and 4 consider Γ_{NN} being the minimizer of (6). Using the same proof technique, one can derive a similar upper bound for Γ_{NN} being the minimizer of (8), up to a constant factor depending on $L_{E_{\mathcal{X}}^n}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}^n}$ and $L_{D_{\mathcal{Y}}^n}$. In the rest of this paper, we only consider the loss function in (6).

In the proof of Theorem 3 and 4, the generalization error is decomposed into a bias term and a variance term. The bias term is bounded using the network approximation

error, and the variance term is bounded using the covering number of the network class, which is closely related with the Rademacher complexity. An alternative argument using local Rademacher complexity Bartlett et al. (2005); Koltchinskii (2006) leads to the same upper bound. Note that a vanilla Rademacher complexity argument will result in a slower convergence rate.

In Theorem 4, we have chosen the optimal $\widetilde{L}\widetilde{p}$ to balance the bias and variance term. For readers who are interested in the generalization error with arbitrary network depth L and width p, please see our proof in Section 7.3. The constants in both theorems only depend on the settings of the problem, and the choices of encoders and decoders. They do not depend on properties of Γ_{NN}. With proper choices of encoders and decoders, such as PCA, our framework is discretization-invariant, see Section 4 and 5 for some popular choices of encoders and decoders.

For a general framework of operator learning using deep neural networks, Theorem 3 and 4 unveils how the generalization error of scales with the number of samples, if the network architecture is properly set. For both network architectures, the upper bound in (11) and (14) consists of a network estimation error and the projection errors in the $\mathcal{X}$ and $\mathcal{Y}$ space.

- The first two terms in (11) and the first term in (14) represent the **network estimation error** for the transformation $\Gamma : \mathbb{R}^{d_\mathcal{X}} \to \mathbb{R}^{d_\mathcal{Y}}$ which maps the encoded vector $E^n_\mathcal{X}(u)$ for u in $\mathcal{X}$ to the encoded vector $E^n_\mathcal{Y}(\Phi(u))$ for $\Phi(u)$ in $\mathcal{Y}$. This error decays exponentially as the sample size n increases with an exponent depending on the dimension $d_\mathcal{X}$ of the encoded space. The dimension $d_\mathcal{X}$ appears in the exponent and $d_\mathcal{Y}$ appears as a constant factor. This is because that the transformation Γ has $d_\mathcal{Y}$ outputs and each output is a function from $\mathbb{R}^\mathcal{X}$ to $\mathbb{R}$. Therefore the rate is only cursed by the input dimension $d_\mathcal{X}$. Note that the minimax rate for learning a $\mathcal{C}^1$ function in $\mathbb{R}^d$ is $n^{-\frac{2}{2+d}}$ (Györfi et al., 2002). Thus for the network estimation error, our rate is optimal up to a logarithmic factor.

- The last two terms in (11) and (14) are **projection errors** in the $\mathcal{X}$ and $\mathcal{Y}$ space, respectively. If the measure γ is concentrated near a $d_\mathcal{X}$-dimensional subspace in $\mathcal{X}$, both projection errors can be made small if the encoder and decoder are properly chosen as the projection onto this $d_\mathcal{X}$-dimensional subspace (see Section 6).

DeepONet (Lu et al., 2021) is another popular framework for learning operators by neural networks. DeepONet uses a subnetwork (the trunk net) to learn a set of functions, and the operator is represented as a linear combination of the trunk nets where the weights are computed by another subnetwork (the branch net). The main differences between DeepONet and the framework studied in this paper are: (i) In DeepONet, the output is given by the dot product between trunk nets and branch nets, while our framework uses standard feedforward neural networks on latent features. (ii) DeepONet uses a basis

decoder for the output space where the bases are given by the trunk nets. The trunk nets are trained together with the branch nets. In our framework, we consider general encoder/decoder, and neural networks are trained to learn the latent transformation. The encoder/decoder and the neural networks are trained separately. The generalization error of DeepONet is analyzed in Lanthaler et al. (2022), with different assumptions from ours. Lanthaler et al. (2021) assumes a Lipscthiz property of the network with respect to the network weight parameters in Lanthaler et al. (2022, Assumption 5.3). This assumption is used to simplify the variance estimation. This assumption is difficult to validate in practical applications, and may not be satisfied. As shown in the proof of Chen et al. (2022, Lemma 5.3 in Appendix C3), without additional conditions, the Lipschitz constant of the network with respect to the network weight parameters scales like p^L where p is the width and L is the depth of the network. In our setting, we do not make any assumption on the network's Lipschitz property with respect to the weight parameters. Instead we use Chen et al. (2022, Lemma 5.3) to bound the network covering number, and further bound the variance in nonparametric estimation. Note that our Lipschitz assumption in Assumption 3 is with respect to the input, and therefore can be easily validated for the linear encoder/decoder.

We next compare the difference between the network architectures in Theorem 3 and Theorem 4. Denote the network architecture in Theorem 3 and Theorem 4 by $\mathcal{F}_1$ and $\mathcal{F}_2$, respectively. The architecture $\mathcal{F}_1$ has the depth and width scaling properly with respect to each other, and an upper bound on all weight parameters and a cardinality constraint. The cardinality constraint is nonconvex and therefore not practical for training this neural network. The architecture $\mathcal{F}_2$ has more flexibility in the choice of depth and width as long as (13) is satisfied. The cardinality constraint is removed for practical concerns. When we set $\widetilde{L} = O(\log n), \widetilde{p} = O(n^{\frac{d_{\mathcal{X}}}{4+2d_{\mathcal{X}}}} \log^{-1} n)$ in $\mathcal{F}_2$, both networks have a depth of $O(\log n)$, while the width of $\mathcal{F}_1$ is the square of that of $\mathcal{F}_2$, i.e., $\mathcal{F}_1$ is wider than $\mathcal{F}_2$. The comparison between $\mathcal{F}_1$ and $\mathcal{F}_2$ is summarized in Table 2.

	$\mathcal{F}_1$ in (2)	$\mathcal{F}_2$ in (3)
General comparison		
Network architecture with a given n	Fixed L and p depending on n	One has the flexibility to choose L and p as long as (13) depending on n is satisfied
Constraints on cardinality	Yes	No
Constraints on the magnitude of weight parameters	Yes	No
Set $\widetilde{L} = O(\log n), \widetilde{p} = O(n^{\frac{d_{\mathcal{X}}}{4+2d_{\mathcal{X}}}} \log^{-1} n)$ in $\mathcal{F}_2$		
Depth L	$O(\log n)$	$O(\log n)$
Width p	$O\left(d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{2+d_{\mathcal{X}}}}\right)$	$O\left(d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{4+2d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{4+2d_{\mathcal{X}}}}\right)$

Table 2: Comparison of the network architectures in Theorem 3 and 4.

In the rest of this paper, we focus on the network architecture in Theorem 4 and discuss its applications in various scenarios. Theorem 3 can also be applied in each case with a similar upper bound.

4. Generalization error with basis encoders and decoders

In this section, we discuss the application of Theorem 4 when the encoder is chosen to be a deterministic basis encoder with a given orthonormal basis of the Hilbert space. Popular choices of orthonormal bases include orthogonal polynomials (e.g., Legendre polynomials (Szeg, 1939; Chkifa et al., 2015; Cohen and DeVore, 2015)) and trigonometric functions (Orszag, 1971; Chen and Shen, 1998; Li et al., 2016).

4.1 Basis encoders and decoders

Let $\mathcal{H}$ be a separable Hilbert space equipped with an inner product $\langle\cdot,\cdot\rangle_{\mathcal{H}}$, and $\{\phi_k\}_{k=1}^{\infty}$ be an orthonormal basis of $\mathcal{H}$ such that $\langle\phi_{k_1},\phi_{k_2}\rangle_{\mathcal{H}}=0$ whenever $k_1\neq k_2$ and $\|\phi_k\|_{\mathcal{H}}=1$ for any k. For any $u\in\mathcal{H}$, we have

$$u=\sum_{k=1}^{\infty}\langle u,\phi_k\rangle_{\mathcal{H}}\phi_k. \tag{15}$$

For a fixed positive integer d representing the encoding dimension, we define the encoder of $\mathcal{H}$ as

$$E_{\mathcal{H},d}(u)=[\langle u,\phi_1\rangle_{\mathcal{H}},...,\langle u,\phi_d\rangle_{\mathcal{H}}]^{\top}\in\mathbb{R}^d,\quad\text{for any } u\in\mathcal{H}, \tag{16}$$

which gives rise to the coefficients associated with a fixed set of d basis functions in the decomposition (15). The decoder $D_{\mathcal{H},d}$ is defined as

$$D_{\mathcal{H},d}(\mathbf{a})=\sum_{k=1}^{d}a_k\phi_k,\ \text{for any } \mathbf{a}\in\mathbb{R}^d. \tag{17}$$

The basis encoder and decoder naturally satisfy the Lipchitz property with a Lipschitz constant 1 (see a proof of Lemma 6 in Appendix B).

Lemma 6 *The encoder $E_{\mathcal{H},d}$ and decoder $D_{\mathcal{H},d}$ defined in (16) and (17) satisfy*

$$\|E_{\mathcal{H},d}(u)-E_{\mathcal{H},d}(\widetilde{u})\|_2\leq\|u-\widetilde{u}\|_{\mathcal{H}}, \tag{18}$$

$$\|D_{\mathcal{H},d}(\mathbf{a})-D_{\mathcal{H},d}(\widetilde{\mathbf{a}})\|_{\mathcal{H}}=\|\mathbf{a}-\widetilde{\mathbf{a}}\|_2, \tag{19}$$

for any $u,\widetilde{u}\in\mathcal{H}$ and $\mathbf{a},\widetilde{\mathbf{a}}\in\mathbb{R}^d$.

Remark 7 *All encoders in the form of (16) are linear operators and therefore satisfy Assumption 5 as long as Assumption 4(ii) holds.*

4.2 Generalization error with basis encoders

We next consider the generalization error when the elements in $\mathcal{X}$ and $\mathcal{Y}$ are encoded by basis encoders with the encoding dimension $d_{\mathcal{X}}$ and $d_{\mathcal{Y}}$, respectively. Substituting the Lipschitz constants of all encoders and decoders by 1 in Theorem 4, we obtain the following corollary:

Corollary 8 *In Setting 1, suppose Assumption 1 – 4 hold. Let* Γ_{NN} *be the minimizer of (6) with the network architecture* $\mathcal{F}(d_{\mathcal{Y}}, L, p, M)$ *in (3) with*

$$L = O(\widetilde{L}),\ p = O\left(\widetilde{p}\right), M = \sqrt{d_{\mathcal{Y}}} R_{\mathcal{Y}}, \tag{20}$$

where $\widetilde{L}, \widetilde{p} > 0$ *are positive integers satisfying* (13)*. Then we have*

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}} \mathbb{E}_{u\sim\gamma} \| D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u) \|_{\mathcal{Y}}^2 \\
&\leq C_4 (\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^2 n + 16 L_{\Psi}^2 \mathbb{E}_{\mathcal{S}} \mathbb{E}_{u\sim\gamma} \| \Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u) - u \|_{\mathcal{X}}^2 \\
&\quad + 2 \mathbb{E}_{\mathcal{S}} \mathbb{E}_{w\sim\Psi_{\#}\gamma} \| \Pi_{\mathcal{Y}, d_{\mathcal{Y}}}^n(w) - w \|_{\mathcal{Y}}^2,
\end{aligned} \tag{21}$$

where C_4 *is a constant depending on* $d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{\Psi}$*.*

Popular choices of orthonormal bases are orthogonal polynomials and trigonometric functions. We next provide an upper bound on the generalization error when Legendre polynomials or trigonometric functions are used for encoding and decoding. In the rest of this section, we assume $\mathcal{X} = \mathcal{Y} = L^2([-1,1]^D)$ with the inner product

$$\langle u_1, u_2 \rangle = \int_{[-1,1]^D} u_1(\mathbf{x}) \overline{u_2(\mathbf{x})} d\mathbf{x}, \tag{22}$$

where $\overline{u_2(\mathbf{x})}$ denotes the complex conjugate of $u_2(\mathbf{x})$.

4.3 Legendre polynomials

On the interval $[-1, 1]$, one-dimensional Legendre polynomials $\{\widetilde{P}_k\}_{k=0}^{\infty}$ are defined recursively as

$$\begin{cases}
\widetilde{P}_0(x) = 1, \\
\widetilde{P}_1(x) = x, \\
\widetilde{P}_{k+1}(x) = \frac{1}{k+1} \left[(2k+1) x \widetilde{P}_k(x) - k \widetilde{P}_{k-1}(x) \right].
\end{cases}$$

The Legendre polynomials satisfy

$$\int_{-1}^{1} \widetilde{P}_k(x) \widetilde{P}_l(x) dx = \frac{2}{2k+1} \delta_{kl},$$

where δ_{kl} is the Kronecker delta which equals to 1 if $k = l$ and equals to 0 otherwise. We define the normalized Legendre polynomials as

$$P_k(x) = \sqrt{\frac{2k+1}{2}}\widetilde{P}_k(x).$$

In the Hilbert space $L^2([-1,1]^D)$, the D-variate normalized Legendre polynomials are defined as

$$\phi^{\mathrm{L}}_{\mathbf{k}} = \prod_{j=1}^{D} P_{k_j}(x_j),$$

where $\mathbf{k} = [k_1 \ \cdots \ k_D]^\top$. The orthonormal basis of Legendre polynomials in $L^2([-1,1]^D)$ is $\{\phi^{\mathrm{L}}_{\mathbf{k}}\}_{\mathbf{k}\in\mathbb{N}_0^D}$.

The encoder with Legendre polynomials can be naturally defined as the expansion coefficients associated with low-order polynomials. Specifically, when $\mathcal{X} = L^2([0,1]^D)$, we fix a positive integer $r_{\mathcal{X}}$ representing the highest degree of the polynomials in each dimension and consider the following set of low-order polynomials

$$\Phi^{\mathrm{L},r_{\mathcal{X}}} := \{\phi^{\mathrm{L}}_{\mathbf{k}} : \|\mathbf{k}\|_\infty \le r_{\mathcal{X}}\}.$$

The encoder $E_{\mathcal{X}}$ and decoder $D_{\mathcal{X}}$ can be defined according to (16) and (17) using the basis functions in $\Phi^{\mathrm{L},r_{\mathcal{X}}}$. In the space $\mathcal{Y} = L^2([0,1]^D)$, the encoder $E_{\mathcal{Y}}$ and decoder $D_{\mathcal{Y}}$ can be defined similarly with basis functions in $\Phi^{\mathrm{L},r_{\mathcal{Y}}}$ for some positive integer $r_{\mathcal{Y}}$.

When Legendre polynomials are used for encoding, the encoding error is guaranteed for regular functions, such as Hölder functions.

Definition 9 (Hölder space) *Let $k \ge 0$ be an integer and $0 < \alpha \le 1$. A function $f : [-1,1]^D \to \mathbb{R}$ belongs to the Hölder space $\mathcal{C}^{k,\alpha}([-1,1]^D)$ if*

$$\|f\|_{\mathcal{C}^{k,\alpha}} := \max_{|\mathbf{k}|\le k} \sup_{\mathbf{x}\in[-1,1]^D} |\partial^{\mathbf{k}} f(\mathbf{x})| + \max_{|\mathbf{k}|=k} \sup_{\mathbf{x}_1\neq\mathbf{x}_2\in[-1,1]^D} \frac{|\partial^{\mathbf{k}} f(\mathbf{x}_1) - \partial^{\mathbf{k}} f(\mathbf{x}_2)|}{\|\mathbf{x}_1 - \mathbf{x}_2\|_2^\alpha} < \infty,$$

where $\partial^{\mathbf{k}} f = \frac{\partial^{|\mathbf{k}|} f}{\partial x_1^{k_1} \partial x_2^{k_2} \cdots \partial x_D^{k_D}}$.

For a given k and α, any functions in $\mathcal{C}^{k,\alpha}([-1,1]^D)$ has continuous partial derivatives up to order k. In particular, $\mathcal{C}^{0,1}([-1,1]^D)$ consists of all Lipschitz functions defined on $[-1,1]^D$.

We assume that the probability measure γ in $\mathcal{X}$ and the pushforward measure $\Psi_{\#}\gamma$ in $\mathcal{Y}$ are supported on subsets of the Hölder space.

Assumption 6 (Hölder input and output) *Let $\mathcal{X} = \mathcal{Y} = L^2([-1,1]^D)$ with the inner product (22). For some integer $k > 0$ and $0 < \alpha \leq 1$, the support of the probability measure γ and the pushforward measure $\Psi_{\#}\gamma$ satisfies*

$$\Omega_{\mathcal{X}} \subset \mathcal{C}^{k,\alpha}([-1,1]^D), \quad \Omega_{\mathcal{Y}} \subset \mathcal{C}^{k,\alpha}([-1,1]^D).$$

There exist $C_{\mathcal{H},\mathcal{X}} > 0$ and $C_{\mathcal{H},\mathcal{Y}} > 0$ such that, for any $u \in \Omega_{\mathcal{X}}$ and $v \in \Omega_{\mathcal{Y}}$

$$\|u\|_{\mathcal{C}^{k,\alpha}} < C_{\mathcal{H},\mathcal{X}}, \quad \|v\|_{\mathcal{C}^{k,\alpha}} < C_{\mathcal{H},\mathcal{Y}}.$$

When Legendre polynomials are used to encode Hölder functions, the generalization error for the operator is given as below:

Corollary 10 *In Setting 1, suppose Assumption 1–6 hold. Denote $s = k + \alpha$. Fix positive integers $d_{\mathcal{X}}$ and $d_{\mathcal{Y}}$ such that $d_{\mathcal{X}}^{1/D}$ and $d_{\mathcal{Y}}^{1/D}$ are integers. Suppose the encoders and decoders are chosen as in (16) and (17) with basis functions $\Phi^{\mathrm{L},d_{\mathcal{X}}^{1/D}}$ and $\Phi^{\mathrm{L},d_{\mathcal{Y}}^{1/D}}$ in $\mathcal{X}$ and $\mathcal{Y}$, respectively. Let Γ_{NN} be the minimizer of (6) or (8) with the network architecture $\mathcal{F}(d_{\mathcal{Y}}, L, p, M)$ in (3) where L, p, M are set as in (20). We have*

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u)\|_{\mathcal{Y}}^2 \\
\leq & C_4(\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^2 n + C_5 L_\Psi^2 d_{\mathcal{X}}^{-\frac{2s}{D}} + C_6 d_{\mathcal{Y}}^{-\frac{2s}{D}}.
\end{aligned}$$

where C_4 depends on $d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_\Psi$, and C_5, C_6 depend on $D, C_{\mathcal{H},\mathcal{X}}, C_{\mathcal{H},\mathcal{Y}}, L_\Psi$.

Corollary 10 is proved in Section 7.4. In Corollary 10, the last two terms represent the projection errors in $\mathcal{X}$ and $\mathcal{Y}$, respectively. When D is large, both terms decay slowly as $d_{\mathcal{X}}$ and $d_{\mathcal{Y}}$ increase. These two error terms remain the same if we choose the encoders given by finite element bases in traditional numerical PDE methods. For example, we consider learning a PDE solver where the operator Ψ represents a map from the initial condition to the PDE solution at a certain time. Assumption 6 assumes that the initial condition and the PDE solution are Hölder functions. Suppose we discretize the domain and represent the solution by finite element basis such that the diameter of all finite elements is no larger than h for some $0 < h < 1$. Let $W^{k,2}([-1,1]^D)$ denote the Sobolev space. We say a set of basis functions are k–order if they are in $W^{k,2}([-1,1]^D)$. If the finite element method with k–th order basis functions is used to approximate the PDE solution, under appropriate assumptions and for any positive integer k, the squared approximation error is $O(h^{2k})$ (Ern and Guermond, 2004, Corollary 1.109). In this case, the total number of basis functions is $O(h^{-D})$. Taking such a finite element approximation as our encoder for $\Omega_{\mathcal{X}}$, we have $d_{\mathcal{X}} = O(h^{-D})$ and the resulting squared projection error is of $O(d_{\mathcal{X}}^{-\frac{2k}{D}})$. In particular, if sparse grids (Bungartz and Griebel, 2004) are used to construct basis functions, the approximation errors for the encoder and decoder can be further reduced.

In the setting of Corollary 10, we only assume the global smoothness of input and output functions. The global approximation encoder by Legendre polynomials (or trigonometric functions in the following subsection) leads to a slow rate of convergence: In Corollary 10, if we choose $d_{\mathcal{X}} = (\log n)^{\frac{1}{2}}$ and when $n \geq \exp\left(\max\left\{100, \left(\frac{7}{2} + \frac{s}{2D}\right)^6\right\}\right)$, the squared generalization error decays in the order of $(\log n)^{-\frac{s}{D}}$ (see a derivation in Appendix A).

However, in practice, when we solve PDEs, the initial conditions and PDE solutions often exhibit low-dimensional structures. For example, the initial conditions and PDE solutions often lie on a low-dimensional subspace or manifold, or the solver itself has low complexity (see Section 6 and Haasdonk (2017); Rozza (2014) for details). Therefore, one can use a few bases (small $d_{\mathcal{X}}$ and $d_{\mathcal{Y}}$) to achieve a small projection error, leading to a fast rate of convergence in the generalization error.

Although using Legendre bases as encoders and decoders requires a uniform sampling of functions, when nonuniform samples are given, one can always use interpolations to generate uniform data and then compute the Legendre coefficients. With this strategy, the whole process is still discretization invariant.

4.4 Trigonometric functions

Trigonometric functions and the Fourier transform have been widely used in various applications where the computation is converted from the spacial domain to the frequency domain. Let $\{T_k(x)\}_{k=1}^{\infty}$ be one-dimensional trigonometric functions defined on $[-1, 1]$ such that

$$\begin{cases} T_1 = 1/2, \\ T_{2k} = \sin(k\pi x) \quad \text{for } k > 1, \\ T_{2k+1} = \cos(k\pi x) \quad \text{for } k > 1. \end{cases} \tag{23}$$

In the Hilbert space $L^2([-1, 1]^D)$, the trigonometric basis is given as $\{\phi_{T,\mathbf{k}}\}_{\mathbf{k}\in\mathbb{N}^D}$ with

$$\phi_{\mathbf{k}}^{\mathrm{T}}(\mathbf{x}) = \prod_{j=1}^{D} T_{k_j}(x_j). \tag{24}$$

When $\mathcal{X} = L^2([0, 1]^D)$, we fix a positive integer $r_{\mathcal{X}}$ and define the set of low-frequency basis

$$\Phi^{\mathrm{T},r_{\mathcal{X}}} = \{\phi_{\mathbf{k}}^{\mathrm{T}} : \|\mathbf{k}\|_\infty \leq r_{\mathcal{X}}\}.$$

We set the encoder $E_{\mathcal{X}}$ and decoder $D_{\mathcal{X}}$ in $\mathcal{X}$ according to (16) and (17) using the basis functions in $\Phi^{\mathrm{T},r_{\mathcal{X}}}$. Similarly, we set the encoder $E_{\mathcal{Y}}$ and decoder $D_{\mathcal{Y}}$ in $\mathcal{Y}$ using the basis functions in $\Phi^{\mathrm{T},r_{\mathcal{Y}}}$ for some positive integer $r_{\mathcal{Y}}$.

Let $\mathcal{P}$ be the set of periodic functions on $[-1, 1]^D$. We assume that the input and output functions are periodic Hölder functions.

Assumption 7 *Let $\mathcal{X} = \mathcal{Y} = L^2([-1,1]^D)$ with the inner product (22). For some integer $k > 0$ and $0 < s \leq 1$, the support of the probability measure γ and the pushforward measure $\Psi_{\#}\gamma$ satisfies*

$$\Omega_{\mathcal{X}} \subset \mathcal{P} \cap \mathcal{C}^{k,\alpha}([-1,1]^D), \quad \Omega_{\mathcal{Y}} \subset \mathcal{P} \cap \mathcal{C}^{k,\alpha}([-1,1]^D).$$

There exist $C_{\mathcal{H}_P,\mathcal{X}} > 0$ and $C_{\mathcal{H}_P,\mathcal{Y}} > 0$ such that for any $u \in \Omega_{\mathcal{X}}$ and $v \in \Omega_{\mathcal{Y}}$

$$\|u\|_{\mathcal{C}^{k,\alpha}} < C_{\mathcal{H}_P,\mathcal{X}}, \quad \|v\|_{\mathcal{C}^{k,\alpha}} < C_{\mathcal{H}_P,\mathcal{Y}}.$$

When trigonometric functions are used to encode periodic Hölder functions, the generalization error for the operator is given as below:

Corollary 11 *Consider Setting 1. Suppose Assumption 1–5 and 7 hold. Denote $s = k + \alpha$. Fix positive integers $d_{\mathcal{X}}$ and $d_{\mathcal{Y}}$ such that $d_{\mathcal{X}}^{1/D}$ and $d_{\mathcal{Y}}^{1/D}$ are integers. Suppose the encoders and decoders are chosen as in (16) and (17) with basis functions $\Phi^{\mathrm{T},d_{\mathcal{X}}^{1/D}}$ and $\Phi^{\mathrm{T},d_{\mathcal{Y}}^{1/D}}$ for $\mathcal{X}$ and $\mathcal{Y}$, respectively. Let Γ_{NN} be the minimizer of (6) or (8) with the network architecture $\mathcal{F}(d_{\mathcal{Y}}, L, p, M)$ in (3) where L, p, M are set as in (20). We have*

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u)\|_{\mathcal{Y}}^2 \\
\leq& C_4(\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^2 n + C_7 L_\Psi^2 d_{\mathcal{X}}^{-\frac{2s}{D}} + C_8 d_{\mathcal{Y}}^{-\frac{2s}{D}}.
\end{aligned}$$

where C_4 depends on $d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_\Psi$, and C_7, C_8 depend on $D, C_{\mathcal{H}_P,\mathcal{X}}, C_{\mathcal{H}_P,\mathcal{Y}}, L_\Psi$.

Corollary 11 is proved in Section 7.5. When using trigonometric functions as encoders and decoders, one can apply Fourier transform when uniform samplings are given and non-uniform Fourier transform when the given samples are non-uniform. The overall framework is discretization invariant. The generalization error with trigonometric basis encoder in Corollary 11 is similar to the error with Legendre polynomials in Corollary 10. If only the global smoothness of input and output functions is assumed, the generalization error decays at a low rate. A faster rate can be achieved if we exploit the low-dimensional structures of the input and output functions.

5. Generalization error for PCA encoders and decoders

When the given data are concentrated near a low-dimensional subspace, PCA is an effective tool for dimension reduction. In this section, we consider the PCA encoder, where the orthonormal basis is estimated from the training data.

5.1 PCA encoders and decoders

Let ρ be a probability measure on a separable Hilbert space $\mathcal{H}$. Define the covariance operator with respect to ρ as

$$G_\rho = \mathbb{E}_{u\sim\rho}[u \otimes u], \tag{25}$$

where $\otimes$ denotes the outer product $(f \otimes g)(h) = \langle g, h\rangle_{\mathcal{H}} f$ for any $f, g, h \in \mathcal{H}$, and $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ denotes the inner product in $\mathcal{H}$. Let $\{\lambda_k\}_{k=1}^{\infty}$ be the eigenvalues of G_ρ in a non-increasing order, and ϕ_k be the eigenfunction associated with λ_k. For any $u \in \mathcal{H}$, we have

$$u = \sum_{j=1}^{\infty} \langle u, \phi_j \rangle_{\mathcal{H}} \phi_j.$$

For a fixed positive integer d, the eigenfunctions $\{\phi_k\}_{k=1}^{d}$ associated with the top d eigenvalues are called the first d principal components. Fixing d, we define the encoder operator $E_{\mathcal{H},d} : \mathcal{H} \to \mathbb{R}^d$ as

$$E_{\mathcal{H},d}(u) = [\langle u, \phi_1 \rangle, \langle u, \phi_2 \rangle, ..., \langle u, \phi_d \rangle]^\top, \text{ for any } u \in \mathcal{H}, \tag{26}$$

which gives rise to the coefficients of u associated with the first d principal components. The decoder $D_{\mathcal{H},d} : \mathbb{R}^d \to \mathcal{H}$ is defined as

$$D_{\mathcal{H},d}(\mathbf{a}) = \sum_{j=1}^{d} a_j \phi_j, \text{ for any } \mathbf{a} = [a_1, ..., a_d]^\top \in \mathbb{R}^d. \tag{27}$$

Given n i.i.d samples $\{u_i\}_{i=1}^{n}$ from ρ, the empirical covariance operator is

$$G_\rho^n = \frac{1}{n} \sum_{i=1}^{n} u_i \otimes u_i. \tag{28}$$

Let $\{\lambda_k^n\}_{k=1}^{\infty}$ be the eigenvalues of G_ρ^n in a non-increasing order, and ϕ_k^n be the eigenfunction associated with λ_k^n. We define the empirical encoder $E_{\mathcal{H},d}^n : \mathcal{H} \to \mathbb{R}^d$ as

$$E_{\mathcal{H},d}^n(u) = [\langle u, \phi_1^n \rangle, \langle u, \phi_2^n \rangle, ..., \langle u, \phi_d^n \rangle]^\top \text{ for any } u \in \mathcal{H}. \tag{29}$$

The empirical decoder is

$$D_{\mathcal{H},d}^n(\mathbf{a}) = \sum_{j=1}^{d} a_j \phi_j^n \text{ for any } \mathbf{a} \in \mathbb{R}^d. \tag{30}$$

The PCA encoders and decoders $E_{\mathcal{H},d}, D_{\mathcal{H},d}, E_{\mathcal{H},d}^n, D_{\mathcal{H},d}^n$ are Lipchitz operators with a Lipchitz constant 1.

Lemma 12 *Let $\mathcal{H}$ be a separable Hilbert space and ρ be a probability measure on $\mathcal{H}$. For any integer $d > 0$, let $E_{\mathcal{H},d}$ and $D_{\mathcal{H},d}$ be the PCA encoder and decoder and $E_{\mathcal{H},d}^n$ and $D_{\mathcal{H},d}^n$ be their empirical counterparts. Then we have*

$$\|E_{\mathcal{H},d}^n(u) - E_{\mathcal{H},d}^n(\widetilde{u})\|_2 \leq \|u - \widetilde{u}\|_{\mathcal{H}}, \text{ for any } u, \widetilde{u} \in \mathcal{H},$$
$$\|D_{\mathcal{H},d}^n(\mathbf{a}) - D_{\mathcal{H},d}^n(\widetilde{\mathbf{a}})\|_{\mathcal{H}} = \|\mathbf{a} - \widetilde{\mathbf{a}}\|_2, \text{ for any } \mathbf{a}, \widetilde{\mathbf{a}} \in \mathbb{R}^d.$$

Lemma 12 can be proved in the same way as Lemma 6. The proof is omitted here.

5.2 Generalization error with PCA encoders and decoders

In this subsection, we choose PCA encoders and decoders for $\mathcal{X}$ and $\mathcal{Y}$. For the $\mathcal{X}$ space, we define the covariance operator and its empirical counterpart as

$$G_\gamma = \mathbb{E}_{u\sim\gamma} u \otimes u \quad \text{and} \quad G_\gamma^n = \frac{1}{n}\sum_{i=1}^n u_i \otimes u_i.$$

Let $\{\phi_{\gamma,k}\}_{k=1}^{d_\mathcal{X}}$ and $\{\phi_{\gamma,k}^n\}_{k=1}^{d_\mathcal{X}}$ be the first $d_\mathcal{X}$ principle components of G_γ and G_γ^n, respectively. The PCA encoder and its empirical counterpart are given as

$$E_\mathcal{X}(u) = [\langle u, \phi_{\gamma,1}\rangle, \langle u, \phi_{\gamma,2}\rangle, ..., \langle u, \phi_{\gamma,d_\mathcal{X}}\rangle]^\top, \ D_\mathcal{X}(\mathbf{a}) = \sum_{j=1}^{d_\mathcal{X}} a_j \phi_{\gamma,j}, \tag{31}$$

$$E_\mathcal{X}^n(u) = \left[\langle u, \phi_{\gamma,1}^n\rangle, \langle u, \phi_{\gamma,2}^n\rangle, ..., \langle u, \phi_{\gamma,d_\mathcal{X}}^n\rangle\right]^\top, \ D_\mathcal{X}^n(\mathbf{a}) = \sum_{j=1}^{d_\mathcal{X}} a_j \phi_{\gamma,j}^n \tag{32}$$

for any $u \in \mathcal{X}$ and $\mathbf{a} \in \mathbb{R}^{d_\mathcal{X}}$.

For the $\mathcal{Y}$ space, the ideal covariance operator in the noiseless case is defined based on the pushforward measure $\Psi_{\#}\gamma$. In the noisy case, the samples $\{v_i\}_{i=1}^n$ are random copies of $\Psi(u) + \widetilde{\epsilon}$. Denote the probability measure of v by ζ. The ideal and empirical covariance operators are defined as

$$G_{\Psi_{\#}\gamma} = \mathbb{E}_{w\sim\Psi_{\#}\gamma} w \otimes w \quad \text{and} \quad G_\zeta^n = \frac{1}{n}\sum_{i=1}^n v_i \otimes v_i.$$

Notice that G_ζ^n is the empirical counterpart of G_ζ, which is different from $G_{\Psi_{\#}\gamma}$ in the noisy case.

Let $\{\phi_{\Psi_{\#}\gamma,k}\}_{k=1}^{d_\mathcal{Y}}$ and $\{\phi_{\zeta,k}^n\}_{k=1}^{d_\mathcal{Y}}$ be the first $d_\mathcal{Y}$ principle components of $G_{\Psi_{\#}\gamma}$ and G_ζ^n, respectively. We choose the PCA encoder:

$$E_\mathcal{Y}(w) = \left[\langle w, \phi_{\Psi_{\#}\gamma,1}\rangle, \langle w, \phi_{\Psi_{\#}\gamma,2}\rangle, ..., \langle w, \phi_{\Psi_{\#}\gamma,d_\mathcal{Y}}\rangle\right]^\top, \ D_\mathcal{X}(\mathbf{a}) = \sum_{j=1}^{d_\mathcal{Y}} a_j \phi_{\Psi_{\#}\gamma,j}, \tag{33}$$

$$E_\mathcal{Y}^n(w) = \left[\langle w, \phi_{\zeta,1}^n\rangle, \langle u, \phi_{\zeta,2}^n\rangle, ..., \langle u, \phi_{\zeta,d_\mathcal{Y}}^n\rangle\right]^\top, \ D_\mathcal{Y}^n(\mathbf{a}) = \sum_{j=1}^{d_\mathcal{Y}} a_j \phi_{\zeta,j}^n \tag{34}$$

for any $w \in \mathcal{Y}$ and $\mathbf{a} \in \mathbb{R}^{d_\mathcal{Y}}$.

The following theorem gives a bound on the generalization error of operator estimation with PCA encoders:

Theorem 13 *In Setting 1, suppose Assumption 1–2 and 4 hold. Consider the PCA encoders and decoders defined in (31)–(34). Let $\{\lambda_k\}_{k=1}^\infty$ be the eigenvalues of the covariance*

operator $G_{\Psi_{\#}\gamma}$ in nonincreasing order. Let Γ_{NN} be the minimizer of (6) with the network architecture $\mathcal{F}(d_{\mathcal{Y}}, L, p, M)$ in (3), where L, p, M are set as in (20). We have

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u)\|_{\mathcal{Y}}^2 \\
\leq & C_4(\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^2 n + 8\left(4R_{\mathcal{X}}^2 L_\Psi^2 \sqrt{d_{\mathcal{X}}} + (R_{\mathcal{Y}} + \widetilde{\sigma})^2 \sqrt{d_{\mathcal{Y}}}\right) n^{-\frac{1}{2}} \\
&+ 16\widetilde{\sigma}^2 \left(\frac{\widetilde{\sigma}}{\lambda_{d_{\mathcal{Y}}} - \lambda_{d_{\mathcal{Y}}+1}}\right)^2 (R_{\mathcal{Y}} + \widetilde{\sigma})^2 + 20\widetilde{\sigma}^2 \\
&+ 16L_\Psi^2 \mathbb{E}_{u\sim\gamma}\|\Pi_{\mathcal{X},d_{\mathcal{X}}}(u) - u\|_2^2 + 16\mathbb{E}_{w\sim\Psi_{\#}\gamma}\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}(w) - w\|_{\mathcal{Y}}^2
\end{aligned} \tag{35}$$

where C_4 is a constant depending on $d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_\Psi$.

Theorem 13 is proved in Section 7.6. Since PCA is discretization invariant, our framework with PCA encoders and decoders enjoys the same desirable property. PCA is effective when the input and output samples are concentrated near low-dimensional subspaces. In this case, an orthonormal basis of the subspace is estimated from the samples. Since the PCA encoder and decoder are data-driven, we expect the corresponding projection errors are smaller than those by Legendre polynomials or trigonometric functions.

In the generalization error in Theorem 13, the error $16\widetilde{\sigma}^2 \left(\frac{\widetilde{\sigma}}{\lambda_{d_{\mathcal{Y}}} - \lambda_{d_{\mathcal{Y}}+1}}\right)^2 (R_{\mathcal{Y}} + \widetilde{\sigma})^2 + 20\widetilde{\sigma}^2$ does not decay as n increases. This is because PCA extracts the principal components from noisy data but does not denoise the data set without additional assumptions on noise. If the noise does not perturb the space spanned by the first $d_{\mathcal{Y}}$ principal eigenfunctions of $G_{\Psi_{\#}\gamma}$, the constant terms can be dropped as the following corollary.

Corollary 14 *Under the conditions of Theorem 13, if the eigenspace spanned by the first $d_{\mathcal{Y}}$ principal eigenfunctions of G_μ coincides with that of $G_{\Psi_{\#}\gamma}$, then we have*

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u)\|_{\mathcal{Y}}^2 \\
\leq & C_4(\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^2 n + 8\left(4R_{\mathcal{X}}^2 L_\Psi^2 \sqrt{d_{\mathcal{X}}} + (R_{\mathcal{Y}} + \widetilde{\sigma})^2 \sqrt{d_{\mathcal{Y}}}\right) n^{-\frac{1}{2}} \\
&+ 16L_\Psi^2 \mathbb{E}_{u\sim\gamma}\|\Pi_{\mathcal{X},d_{\mathcal{X}}}(u) - u\|_2^2 + 16\mathbb{E}_{w\sim\Psi_{\#}\gamma}\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}(w) - w\|_{\mathcal{Y}}^2.
\end{aligned} \tag{36}$$

Corollary 14 is proved in Section 7.7.

6. Exploit additional low-dimensional structures

Section 4 and Section 5 are suitable for the case where the input and output samples are concentrated near a low-dimensional subspace. While in practice, the low-dimensional subspace is not a priori known. In order to capture such a subspace, we need to choose a large encoding dimension so that the low-dimensional subspace is enclosed by the encoded space, which guarantees a small projection error. However, the network estimation error

(see Section 3.2 for the definition) has an exponential dependence on $d_{\mathcal{X}}$. The error decays slowly when $d_{\mathcal{X}}$ is large.

Additionally, the given data may be located on a low-dimensional manifold enclosed by the encoded space, or the operator Ψ may have low complexity. In this section, we will exploit such additional low-dimensional structures. We will show that, even though $d_{\mathcal{X}}$ and $d_{\mathcal{Y}}$ are chosen to be large in order to guarantee small projection errors, the exponent in the network estimation error only depends on the intrinsic dimension of the additional low-dimensional structures of data, instead of $d_{\mathcal{X}}$. Specifically, we consider two scenarios : (1) when the collection of encoded vectors $E_{\mathcal{X}}(\Omega_{\mathcal{X}})$ is on a low-dimensional manifold and (2) when the operator Ψ only depends on a few directions in the encoded space.

6.1 When encoded vectors lie on a low-dimensional manifold

We first consider the case when the given data exhibit a nonlinear low-dimensional structure: For a given encoder $E_{\mathcal{X}} : \mathcal{X} \to \mathbb{R}^{d_{\mathcal{X}}}$, the encoded vectors $\{E_{\mathcal{X}}(u) : u \text{ is randomly sampled from } \gamma\}$ lie on a d_0-dimensional manifold with $d_0 \ll d_{\mathcal{X}}$. This scenario is observed in many applications. For example, the solutions of most PDEs are in an infinite-dimensional function space. After uniform discretization, the solutions are encoded to vectors in a very high dimensional space. For many PDEs, it is commonly observed that the solutions actually lie on a low-dimensional manifold enclosed by the discretized high-dimensional space. Therefore the solution manifold can be well-approximated using much fewer bases than those used in the discretization. This observation leads to the success of the reduced basis method (Haasdonk, 2017; Rozza, 2014). Another concrete example is described as follows:

Example 1 *Let $\mathcal{X} = L^2([-1,1])$ and $d_0, d_{\mathcal{X}}$ be positive integers such that $d_0 < d_{\mathcal{X}}$. Let $\{T_k\}_{k=1}^{\infty}$ be the trigonometric functions defined in (23) and $\{g_k\}_{k=d_0+1}^{d_{\mathcal{X}}}$ be some real valued functions. Suppose the probability measure γ is supported on*

$$\Omega_{\mathcal{X}} = \left\{ u : u = \sum_{k=1}^{d_{\mathcal{X}}} a_k T_k \text{ with } a_k \in \mathbb{R} \text{ for } k = 1, \ldots, d_0, \text{ and } a_k = g_k(a_1, ..., a_{d_0}) \text{ for } k = d_0+1, \ldots, d_{\mathcal{X}} \right\}.$$

The support set $\Omega_{\mathcal{X}}$ has an intrinsic dimension d_0. If we choose the basis encoder $E_{\mathcal{X}} : \mathcal{X} \to \mathbb{R}^{d_{\mathcal{X}}}$ using the trigonometric functions $\{T_k\}_{k=1}^{d_{\mathcal{X}}}$, then the encoded vectors $\{E_{\mathcal{X}}(u) : u \text{ is randomly sampled from} \gamma\}$ lie on a d_0-dimensional manifold embedded in $\mathbb{R}^{d_{\mathcal{X}}}$. Figure 1 shows this manifold when $d_{\mathcal{X}} = 3, d_0 = 2$ and $g_3 = a_1^2 + a_2$.

This nonlinear low-dimensional structure of data can be described as follows:

Assumption 8 *Let $d_0, d_{\mathcal{X}}$ be positive integers such that $d_0 < d_{\mathcal{X}}$. In Setting 1, there exists an encoder $E_{\mathcal{X}} : \mathcal{X} \to \mathbb{R}^{d_{\mathcal{X}}}$ such that the encoded vectors $\{E_{\mathcal{X}}(u) : u \text{ is randomly sampled from } \gamma\}$ is on a d_0-dimensional compact smooth Riemannian manifold $\mathcal{M}$ isometrically embedded in $\mathbb{R}^{d_{\mathcal{X}}}$. The reach of $\mathcal{M}$ (Federer, 1959; Niyogi et al., 2008) is $\tau > 0$.*

Under Assumption 8 and Setting 1, the output $\Psi(u)$ is perturbed by noise, while the input u is clean and its encoded vector is located on $\mathcal{M}$. Such a setting is common in practice when a series of experiments is conducted to simulate a scientific phenomenon. In experiments, one designs the inputs and takes measurements of the outputs. Usually, the inputs are generated according to some physical laws that lead to low-dimensional structures. Due to the limitations of sensors and equipment, the measured outputs are perturbed by noise.

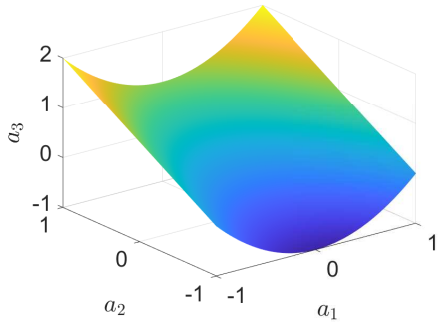

Figure 1: An illustration of Example 1 with $d_{\mathcal{X}} = 3, d_0 = 2$ and $g_3 = a_1^2 + a_2$.

Approximation and statistical estimation theories of deep neural networks for functions on a low-dimensional manifold have been studied in Chen et al. (2019, 2022, 2020); Liu et al. (2021); Shen et al. (2020); Jiao et al. (2021); Cloninger and Klock (2020); Shaham et al. (2018); Schmidt-Hieber (2019); Du et al. (2021); Nakada and Imaizumi (2020b). In this subsection, we show that deep neural networks can automatically adapt to nonlinear low-dimensional structures of data, and give rise to a sample complexity depending on the intrinsic dimension d_0. The following theorem gives a generalization error in this scenario.

Theorem 15 *In Setting 1, suppose Assumption 1–5 and 8 hold, and the encoder $E_{\mathcal{X}}$ in Assumption 8 is given. Let Γ_{NN} be the minimizer of (6) with the network architecture $\mathcal{F}(d_{\mathcal{Y}}, L, p, M)$ in (3) with*

$$L = O(\widetilde{L}),\ p = O\left(d_{\mathcal{X}}\widetilde{p}\right), M = \sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}, \tag{37}$$

where $\widetilde{L}, \widetilde{p} > 0$ are positive integers satisfying

$$\widetilde{L}\widetilde{p} = \left\lceil d_{\mathcal{Y}}^{-\frac{d_0}{4+2d_0}} n^{\frac{d_0}{4+2d_0}} \right\rceil. \tag{38}$$

Then we have

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u)\|_{\mathcal{Y}}^2 \\
\leq\ & C_5(\widetilde{\sigma}^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4+d_0}{2+d_0}} d_{\mathcal{X}}^2 n^{-\frac{2}{2+d_0}} \log^2 n \\
&+ C_3\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|\Pi_{\mathcal{X},d_{\mathcal{X}}}(u) - u\|_{\mathcal{X}}^2 + 2\mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}^n(w) - w\|_{\mathcal{Y}}^2,
\end{aligned} \tag{39}$$

where C_5 dpends on $d_0, \log d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{E_{\mathcal{X}}^n}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}}, L_{D_{\mathcal{Y}}^n}, L_\Psi, \tau$, the surface area of $\mathcal{M}$, and $C_3 = 16 L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_\Psi^2$.

Theorem 15 is proved in Section 7.8. The convergence rate in Theorem 15 has an exponential dependence on d_0, instead of $d_{\mathcal{X}}$. Theorem 15 shows that when the encoded vectors are located on a low-dimensional manifold, deep neural networks are adaptive to such nonlinear geometric structures of data.

6.2 When the operator Ψ has low complexity

In our framework, learning Ψ is converted to learning the transformation $\Gamma : \mathbb{R}^{d_{\mathcal{X}}} \to \mathbb{R}^{d_{\mathcal{Y}}}$, as defined in (5). The second scenario we consider in this subsection is that, even though the u_i's and v_i's are in infinite-dimensional spaces, the operator Ψ has low complexity: its corresponding transformation Γ can be approximated by some low-dimensional functions that only depend on few directions in $\mathbb{R}^{d_{\mathcal{X}}}$. For example, consider solving a linear PDE with constant coefficients by the Fourier spectral method. In this case, the operator Ψ is the PDE solver that maps initial conditions to solutions at certain time. By taking the Fourier transform on both sides of the PDE, solving the PDEs is converted to solving a series of independent ODEs, each of which controls the evolution of a Fourier coefficient of the solution (Shen et al., 2011, Chapter 2). The operator Ψ can be fully characterized by a system of one-dimensional ODEs. We next adapt this setting to our framework in order to learn Ψ. We use trigonometric functions as our encoders and decoders: the initial conditions and solutions are approximated by the first $d_{\mathcal{X}} = d_{\mathcal{Y}}$ terms of their Fourier series expansion. Then learning Ψ reduces to learning $d_{\mathcal{Y}}$ one-dimensional functions, each of which corresponds to an ODE of a Fourier coefficient, instead of learning $d_{\mathcal{Y}}$ $d_{\mathcal{X}}$-dimensional functions.

In this subsection, we show that we can get a faster rate by exploiting the low complexity of Ψ. We first make an assumption on Ψ:

Assumption 9 *Let $0 < d_0 \leq d_{\mathcal{X}}$ be integers. Assume there exist $E_{\mathcal{X}}, D_{\mathcal{X}}, E_{\mathcal{Y}}, D_{\mathcal{Y}}$ such that for any $u \in \Omega_{\mathcal{X}}$, we have*

$$\Pi_{\mathcal{Y},d_{\mathcal{Y}}} \circ \Psi(u) = D_{\mathcal{Y}} \circ \mathbf{g} \circ E_{\mathcal{X}}(u) \tag{40}$$

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

$$\mathbf{g}(\mathbf{a}) = \left[g_1(V_1^\top \mathbf{a}) \quad \cdots \quad g_{d_{\mathcal{Y}}}(V_{d_{\mathcal{Y}}}^\top \mathbf{a}) \right]^\top, \tag{41}$$

for some unknown matrix $V_k \in \mathbb{R}^{d_{\mathcal{X}} \times d_0}$, and some unknown real valued function $g_k : \mathbb{R}^{d_0} \to \mathbb{R}$ where $k = 1, ..., d_{\mathcal{Y}}$.

In statistics, the functions g_k's in Assumption 9 are known as single-index models for $d_0 = 1$, and are known as multi-index models for $d_0 > 1$. For any given $u \sim \gamma$, we decompose $\Psi(u)$ into two parts: the first part is its projection to the set of encoded vectors $E_{\mathcal{Y}}(\Omega_{\mathcal{Y}})$; the second part is the rest orthogonal to the first part. Assumption 9 assumes that the

operator mapping u to the first part follows a multi-index model. When $d_{\mathcal{Y}}$ is large enough, the second part has a small magnitude and is included in the projection error. In the following example, we give a simple illustration when the second part vanishes.

Example 2 *Let $\mathcal{X} = L^2([-1,1])$, $\Omega_{\mathcal{X}} \subset \mathcal{X}$ be a compact set in $\mathcal{P} \cap \mathcal{X}$ and $0 < d_0 < d_{\mathcal{X}}$ be integers. Let $\{T_k\}_{k=1}^{\infty}$ be trigonometric functions defined in (23). Any $u \in \Omega_{\mathcal{X}}$ can be written as $u = \sum_{k=1}^{\infty} a_k T_k$ for some a_k's. Denote $\mathbf{a}_u = \begin{bmatrix} a_1 & \cdots & a_{d_{\mathcal{X}}} \end{bmatrix}^\top$. Suppose the operator we want to learn has the following form*

$$\Psi(u) = \sum_{k=1}^{d_{\mathcal{Y}}} g_k(V_k^\top \mathbf{a}_u) T_k, \tag{42}$$

with $V_k \in \mathbb{R}^{d_{\mathcal{X}} \times d_0}$ and $g_k : \mathbb{R}^{d_0} \to \mathbb{R}$ for $k = 1, ..., d_{\mathcal{Y}}$. We set $E_{\mathcal{X}}, D_{\mathcal{X}}$ as the basis encoder and decoder using the basis functions $\{T_k\}_{k=1}^{d_{\mathcal{X}}}$, and $E_{\mathcal{Y}}, D_{\mathcal{Y}}$ as encoder and decoder derived using basis $\{T_k\}_{k=1}^{d_{\mathcal{Y}}}$. In this example, $\Pi_{\mathcal{Y},d_{\mathcal{Y}}} \circ \Psi(u) = \Psi(u)$ for any $u \sim \gamma$. Then learning Ψ reduces to learning the g_k's and the V_k's. An illustration of the estimator is shown in Figure 2. In neural networks, the V_k's can be realized by a single layer. Therefore, our major task is to learn good approximations of the g_k's. Note that each g_k is a d_0-dimensional function. By exploiting such low complexity of the operator, we can convert the learning task from learning $d_{\mathcal{Y}}$ $d_{\mathcal{X}}$-dimensional functions to learning $d_{\mathcal{Y}}$ d_0-dimensional functions.

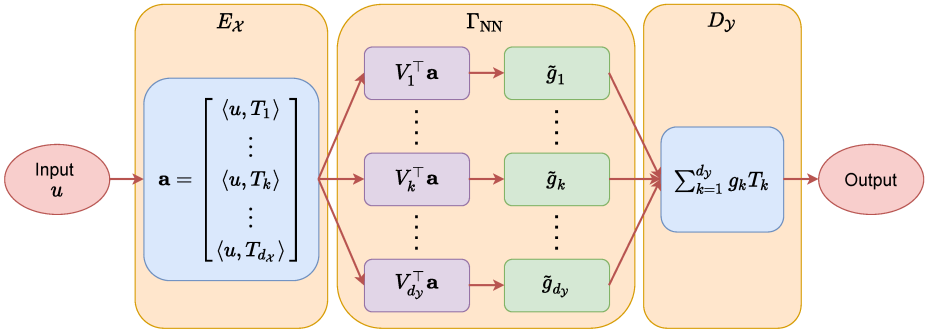

Figure 2: An illustration of Example 2, where the $\widetilde{g}_k$'s represent network approximations of the g_k's in (42).

With Assumption 9, the following theorem gives a faster rate on the generalization error:

Theorem 16 *In Setting 1, suppose Assumption 1–5 and 9 hold. Assume that the encoders and decoders $E_{\mathcal{X}}, D_{\mathcal{X}}, E_{\mathcal{Y}}, D_{\mathcal{Y}}$ in Assumption 9 are given. Let Γ_{NN} be the minimizer of (6) with the network architecture $\mathcal{F}(d_{\mathcal{Y}}, L, p, M)$ in (3), where*

$$L = O(\widetilde{L}),\ p = O\left(\widetilde{p}\right), M = \sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}} \tag{43}$$

and $\widetilde{L}, \widetilde{p} > 0$ *are integers and satisfy (38).*

We have

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D^n_{\mathcal{Y}} \circ \Gamma_{\mathrm{NN}} \circ E^n_{\mathcal{X}}(u) - \Psi(u)\|^2_{\mathcal{Y}} \\
\leq & C_6(\widetilde{\sigma}^2 + R^2_{\mathcal{Y}}) d_{\mathcal{Y}}^{\frac{4+d_0}{2+d_0}} \max\left\{ n^{-\frac{2}{2+d_0}}, d_{\mathcal{X}} n^{-\frac{4+d_0}{4+2d_0}} \right\} \log^2 n \\
&+ C_3 \mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|\Pi_{\mathcal{X},d_{\mathcal{X}}}(u) - u\|^2_{\mathcal{X}} + 2\mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}(w) - w\|^2_{\mathcal{Y}}, \qquad (44)
\end{aligned}$$

where C_6 *depends on* $d_0, \log d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{E^n_{\mathcal{X}}}, L_{E^n_{\mathcal{Y}}}, L_{D^n_{\mathcal{X}}}, L_{D^n_{\mathcal{Y}}}, L_\Psi$, *and* $C_3 = 16 L^2_{D^n_{\mathcal{Y}}} L^2_{E^n_{\mathcal{Y}}} L^2_\Psi$.

Theorem 16 is proved in Section 7.9. In Assumption 9, each V_k is a linear transformation that can be realized by a singly layer. In our network construction, the first layer is used to learn these transformations and the rest is used to learn the functions g_k's.

In Assumption 9, the function $\mathbf{g}$ is a vector-valued function whose elements are multi-index models. Our result in Theorem 16 can be easily extended to the case when $\mathbf{g}$ is a composition of several multi-index models. Specifically, for some $m > 0$, consider $\mathbf{g}$ in the following form

$$\mathbf{g} = \mathbf{g}_m \circ \mathbf{g}_{m-1} \circ \cdots \circ \mathbf{g}_2 \circ \mathbf{g}_1, \qquad (45)$$

where $\mathbf{g}_k : \mathbb{R}^{d_{k-1}} \to \mathbb{R}^{d_k}$ is in the form of

$$\mathbf{g}_k(\mathbf{a}) = \left[g_{k,1}(V_{k,1}^\top \mathbf{a}) \quad \cdots \quad g_{d_k}(V_{k,d_{k-1}}^\top \mathbf{a}) \right]^\top, \qquad (46)$$

for some unknown matrix $V_{k,j} \in \mathbb{R}^{d_{k-1}\times \widetilde{d}_k}$, unknown function $\mathbf{g}_{k,j} : \mathbb{R}^{\widetilde{d}_k} \to \mathbb{R}$ and $d_k, \widetilde{d}_k > 0$ being integers. Here d_k is the dimension of $\mathbf{g}_k$, and $\widetilde{d}_k$ is the dimension of $V_{k,j}\mathbf{a}$. Replace $\mathbf{g}$ in Assumption 9 by the one defined above. The upper bound in Theorem 16 also holds up to a factor depending on $\max_k d_k$ and $\max_k \widetilde{d}_k$.

7. Proof of main results

In this section, we give proofs to our main theorems and corollaries.

7.1 Preliminaries

In this section, we define several quantities that will be used in the proof. We first define two types of covering number of function classes. The first type is independent of data and will be used to prove Theorem 3.

Definition 17 (Cover) *Let* $\mathcal{F}$ *be a class of functions. A set of functions* $\mathcal{S}$ *is a* δ*-cover of* $\mathcal{F}$ *with respect to a norm* $\|\cdot\|$ *if for any* $f \in F$, *one has*

$$\inf_{f^* \in \mathcal{S}} \|f - f^*\| \leq \delta.$$

Definition 18 (Covering number, Definition 2.1.5 of (Van Der Vaart et al., 1996)) *Let F be a class of functions. For any $\delta > 0$, the covering number of $\mathcal{F}$ is defined as*

$$\mathcal{N}(\delta, \mathcal{F}, \|\cdot\|) = \min\{|\mathcal{S}_f| : \mathcal{S}_f \text{ is a } \delta\text{-cover of } \mathcal{F}\},$$

where $|\mathcal{S}_f|$ denotes the cardinality of $\mathcal{S}_f$.

Definition 17 and 18 depend on the norm $\|\cdot\|$. In the following, we choose $\|\cdot\|$ as a sample dependent norm and define the so-called uniform covering number. We first define the cover with respect to samples:

Definition 19 (Cover with respect to samples) *Let $\mathcal{F}$ be a class of functions from $\mathbb{R}^{d_1}$ to $\mathbb{R}^{d_2}$. Given a set of samples $X = \{\mathbf{x}_k\}_{k=1}^m \subset \mathbb{R}^{d_1}$, for any $\delta > 0$, a function set $\mathcal{S}_f(X)$ is a δ-cover of F with respect to X if for any $f \in \mathcal{F}$, there exists $f^* \in \mathcal{S}_f(X)$ such that*

$$\|f(\mathbf{x}_k) - f^*(\mathbf{x}_k)\|_\infty \leq \delta, \quad \forall 1 \leq k \leq m.$$

Definition 19 is a special case of Definition 17 in which the norm $\|\cdot\|$ is chosen as the ℓ^∞ norm of the collection of its argument's values over samples X. Based on Definition 19, we define the uniform covering number as follows:

Definition 20 (Uniform covering number, Section 10.2 of Anthony and Bartlett (1999)) *Let $\mathcal{F}$ be a class of functions from $\mathbb{R}^d$ to $\mathbb{R}$. For any set of samples $X = \{\mathbf{x}_k\}_{k=1}^m \subset \mathbb{R}^d$, denote*

$$\mathcal{F}|_X = \{(f(\mathbf{x}_1), ..., f(\mathbf{x}_m)) : f \in \mathcal{F}\}.$$

For any $\delta > 0$, the uniform covering number of $\mathcal{F}$ with m samples is defined as

$$\mathcal{N}(\delta, \mathcal{F}, m) = \max_{X \subset \mathbb{R}^d, |X|=m} \min_{\mathcal{S}_f(X)} \{|\mathcal{S}_f(X)| : \mathcal{S}_f(X) \text{ is a } \delta\text{-cover of } \mathcal{F} \text{ with respect to } X\}. \tag{47}$$

This covering number is used to prove Theorem 4.

7.2 Proof of Theorem 3

To prove Theorem 3, we first decompose the squared L^2 error $\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\|D_{\mathcal{Y}}^n \circ \Gamma_{\text{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u)\|_{\mathcal{Y}}^2$ into a network estimation error and a projection error. The network estimation error can be further decomposed into a bias term and a variance term. The bias term heavily depends on the approximation error of the network class (2). The variance term is upper bounded in terms of the covering number of the network class.

Proof of Theorem 3. We first decompose the squared L^2 error as

$$\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\left\|D_{\mathcal{Y}}^n \circ \Gamma_{\text{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u)\right\|_{\mathcal{Y}}^2\right]$$

$$\leq \underbrace{2\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\left\|D^n_{\mathcal{Y}}\circ\Gamma_{\mathrm{NN}}\circ E^n_{\mathcal{X}}(u)-D^n_{\mathcal{Y}}\circ E^n_{\mathcal{Y}}\circ\Psi(u)\right\|^2_{\mathcal{Y}}\right]}_{\mathrm{I}}+\underbrace{2\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\left\|D^n_{\mathcal{Y}}\circ E^n_{\mathcal{Y}}\circ\Psi(u)-\Psi(u)\right\|^2_{\mathcal{Y}}\right]}_{\mathrm{II}}. \tag{48}$$

Here I is the network estimation error in the $\mathcal{Y}$ space, II is the empirical projection error, which can be rewritten as

$$\mathrm{II}=2\mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi^n_{\mathcal{Y},d_{\mathcal{Y}}}(w)-w\right\|^2_{\mathcal{Y}}\right]. \tag{49}$$

In the remaining of this subsection, we derive an upper bound of I. Note that I can be bounded as

$$\begin{aligned}\mathrm{I}=&2\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\left\|D^n_{\mathcal{Y}}\circ\Gamma_{\mathrm{NN}}\circ E^n_{\mathcal{X}}(u)-D^n_{\mathcal{Y}}\circ E^n_{\mathcal{Y}}\circ\Psi(u)\right\|^2_{\mathcal{Y}}\right]\\ \leq&2L^2_{D^n_{\mathcal{Y}}}\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\left\|\Gamma_{\mathrm{NN}}\circ E^n_{\mathcal{X}}(u)-E^n_{\mathcal{Y}}\circ\Psi(u)\right\|^2_2\right].\end{aligned} \tag{50}$$

If the training samples in $\mathcal{S}_1$ are fixed, we have the following conditioned on $\mathcal{S}_1$:

$$\begin{aligned}&\mathbb{E}_{\mathcal{S}_2}\mathbb{E}_{u\sim\gamma}\left[\left\|\Gamma_{\mathrm{NN}}\circ E^n_{\mathcal{X}}(u)-E^n_{\mathcal{Y}}\circ\Psi(u)\right\|^2_2\right]\\ =&\underbrace{2\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\|\Gamma_{\mathrm{NN}}\circ E^n_{\mathcal{X}}(u_i)-E^n_{\mathcal{Y}}\circ\Psi(u_i)\right\|^2_2\right]}_{\mathrm{T}_1}\\ &+\underbrace{\mathbb{E}_{\mathcal{S}_2}\mathbb{E}_{u\sim\gamma}\left[\left\|\Gamma_{\mathrm{NN}}\circ E^n_{\mathcal{X}}(u)-E^n_{\mathcal{Y}}\circ\Psi(u)\right\|^2_2\right]-\mathbb{E}_{\mathcal{S}_2}\left[\frac{2}{n}\sum_{i=n+1}^{2n}\left\|\Gamma_{\mathrm{NN}}\circ E^n_{\mathcal{X}}(u_i)-E^n_{\mathcal{Y}}\circ\Psi(u_i)\right\|^2_2\right]}_{\mathrm{T}_2}.\end{aligned} \tag{51}$$

In the decomposition of (51), the term T_1 consists of the bias of using neural network to approximate the transformation Γ and the projection error of $\Pi^n_{\mathcal{X},d_{\mathcal{X}}}$ in the $\mathcal{X}$ space. The term T_2 captures the variance. We next derive bounds for T_1 and T_2 respectively.

Upper bound of T_1. The term T_1 is the expected mean squared error of the learned transformation Γ_{NN} with respect to $\mathcal{S}_2$. We will derive an upper bound using the network approximation error and network architecture's covering number. The network approximation error is the bias. We use network architecture's covering number to bound the stochastic error.

Define the transformation $\Gamma^n_d:\mathbb{R}^{d_{\mathcal{X}}}\to\mathbb{R}^{d_{\mathcal{Y}}}$

$$\Gamma^n_d=E^n_{\mathcal{Y}}\circ\Psi\circ D^n_{\mathcal{X}}, \tag{52}$$

which maps the encoded vector $E^n_{\mathcal{X}}(u)$ in $\mathcal{X}$ to the encoded vector $E^n_{\mathcal{Y}}(v)$ in $\mathcal{Y}$. The transformation Γ^n_d is the target transformation to be estimated by Γ_{NN}. It is straightforward to show that Γ^n_d is a Lipschitz transformation (see a proof of Lemma 21 in Appendix C).

Lemma 21 *Assume Assumption 2 and 3. Γ_d^n is Lipschitz with a Lipschitz constant $L_{E_{\mathcal{Y}}^n} L_{D_{\mathcal{X}}^n} L_\Psi$.*

Denote

$$\boldsymbol{\epsilon}_i = E_{\mathcal{Y}}^n(v_i) - E_{\mathcal{Y}}^n(\Psi(u_i)). \tag{53}$$

According to Assumption 3 and Assumption 4(iii)–(iv), we have

$$\mathbb{E}[\boldsymbol{\epsilon}_i] = \mathbf{0}, \text{ and } \|\boldsymbol{\epsilon}_i\|_\infty < \sigma.$$

We decompose T_1 as

$$\begin{aligned}
\mathrm{T}_1 =& 2\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n\circ\Psi(u_i)\right\|_2^2\right] \\
=& 2\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n\circ\Psi(u_i) - \boldsymbol{\epsilon}_i + \boldsymbol{\epsilon}_i\right\|_2^2\right] \\
=& 2\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n\circ\Psi(u_i) - \boldsymbol{\epsilon}_i\right\|_2^2\right] \\
&+ 4\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\langle\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n\circ\Psi(u_i) - \boldsymbol{\epsilon}_i, \boldsymbol{\epsilon}_i\right\rangle\right] + 2\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\|\boldsymbol{\epsilon}_i\|_2^2\right] \\
=& 2\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n(v_i)\right\|_2^2\right] \\
&+ 4\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\langle\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i), \boldsymbol{\epsilon}_i\right\rangle\right] - 2\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\|\boldsymbol{\epsilon}_i\|_2^2\right] \\
=& 2\mathbb{E}_{\mathcal{S}_2}\left[\inf_{\Gamma\in\mathcal{F}_{\mathrm{NN}}}\frac{1}{n}\sum_{i=n+1}^{2n}\left\|\Gamma\circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n(v_i)\right\|_2^2\right] \\
&+ 4\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\langle\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i), \boldsymbol{\epsilon}_i\right\rangle\right] - 2\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\|\boldsymbol{\epsilon}_i\|_2^2\right] \\
& \qquad\qquad \text{by the definition of } \Gamma_{\mathrm{NN}} \text{ in (6)} \\
\le& 2\inf_{\Gamma\in\mathcal{F}_{\mathrm{NN}}}\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\|\Gamma\circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n(v_i)\right\|_2^2\right] + 4\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\langle\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i), \boldsymbol{\epsilon}_i\right\rangle\right] \\
&- 2\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\|\boldsymbol{\epsilon}_i\|_2^2\right] \\
=& 2\inf_{\Gamma\in\mathcal{F}_{\mathrm{NN}}}\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left[\|\Gamma\circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n\circ\Psi(u_i) - \boldsymbol{\epsilon}_i\|_2^2 - \|\boldsymbol{\epsilon}_i\|_2^2\right]\right] \\
&+ 4\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\left\langle\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i), \boldsymbol{\epsilon}_i\right\rangle\right]
\end{aligned}$$

$$=2\inf_{\Gamma\in\mathcal{F}_{\rm NN}}\mathbb{E}_{u\sim\gamma}\left[\left\|\Gamma\circ E_{\mathcal{X}}^n(u)-E_{\mathcal{Y}}^n\circ\Psi(u)\right\|_2^2\right]+4\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\langle\Gamma_{\rm NN}\circ E_{\mathcal{X}}^n(u_i),\boldsymbol{\epsilon}_i\rangle\right]. \tag{54}$$

In (54), the first term is the neural network approximation error, and the second term is the stochastic error from noise. To derive an upper bound of the first term, we use the following lemma which shows that for any function f in the Sobolev space $W^{k,\infty}$, when the network architecture is properly set, FNN can approximate f with arbitrary accuracy:

Lemma 22 (Theorem 1 of (Yarotsky, 2017)) *Let $k\geq 0$ be a positive integer . There exists an FNN architecture $\mathcal{F}_{\rm NN}(1,L,p,K,\kappa,M)$ capable of approximating any function in $W^{k,\infty}\left([-B,B]^d\right)$, i.e., for any given $\epsilon\in(0,1)$ and if $f\in W^{k,\infty}\left([-B,B]^d\right)$, the network architecture gives rise to a function $\widetilde{f}$ satisfying*

$$\left\|\widetilde{f}-f\right\|_\infty\leq\varepsilon.$$

The hyperparameters in $\mathcal{F}_{\rm NN}$ are chosen as

$$L=O\left(\log\frac{1}{\varepsilon}\right),\ p=O\left(\varepsilon^{-\frac{d}{k}}\right),\ K=O\left(\varepsilon^{-\frac{d}{k}}\log\frac{1}{\varepsilon}\right),\ \kappa=\max\left\{1,B,R\right\},\ M=R.$$

The constant hidden in $O(\cdot)$ depends on k,α,d,B,R.

Remark 23 *Lemma 22 is a variant of (Yarotsky, 2017, Theorem 1). In (Yarotsky, 2017, Theorem 1), it is required that the input is in $[0,1]^D$. For any input in $[-B,B]^D$, one can always rescale and shift the input to $[0,1]^D$ and apply (Yarotsky, 2017, Theorem 1). Such a transformation only affect the Sobolev norm of the target function and the upper bound of weight parameters of $\mathcal{F}_{\rm NN}$. The statement of Lemma 22 has already incorporated such a transformation.*

Since Γ_d^n is Lipschitz by Lemma 21, according to Lemma 22 with $k=1$, for any $\varepsilon_1>0$, there is a network architecture $\mathcal{F}_{\rm NN}(d_{\mathcal{Y}},L,p,K,\kappa,M)$, such that for any Γ_d^n defined in (52), there exists a $\widetilde{\Gamma}_d^n\in\mathcal{F}_{\rm NN}(d_{\mathcal{Y}},L,p,K,\kappa,M)$ with

$$\left\|\widetilde{\Gamma}_d^n-\Gamma_d^n\right\|_\infty\leq\varepsilon_1.$$

Such a network architecture has

$$\begin{aligned}&L=O(\log\varepsilon_1),\ p=O\left(\varepsilon_1^{-d_{\mathcal{X}}}\right),\ K=O\left(\varepsilon_1^{-d_{\mathcal{X}}}\log\varepsilon_1\right),\\&\kappa=\max\left\{1,\sqrt{d_{\mathcal{Y}}}L_{E_{\mathcal{Y}}^n}R_{\mathcal{Y}},\sqrt{d_{\mathcal{X}}}L_{E_{\mathcal{X}}^n}R_{\mathcal{X}},L_{E_{\mathcal{Y}}^n}L_{D_{\mathcal{X}}^n}L_\Psi\right\},\ M=\sqrt{d_{\mathcal{Y}}}L_{E_{\mathcal{Y}}^n}R_{\mathcal{Y}}.\end{aligned} \tag{55}$$

We bound the first term in (54) as

$$\inf_{\Gamma\in\mathcal{F}_{\rm NN}}\mathbb{E}_{u\sim\gamma}\left[\left\|\Gamma\circ E_{\mathcal{X}}^n(u)-E_{\mathcal{Y}}^n\circ\Psi(u)\right\|_2^2\right]$$

$$
\begin{aligned}
&\leq \mathbb{E}_{u\sim\gamma}\left[\left\|\widetilde{\Gamma}_d^n \circ E_{\mathcal{X}}^n(u) - E_{\mathcal{Y}}^n \circ \Psi(u)\right\|_2^2\right] \\
&\leq 2\mathbb{E}_{u\sim\gamma}\left[\left\|\widetilde{\Gamma}_d^n \circ E_{\mathcal{X}}^n(u) - \Gamma_d \circ E_{\mathcal{X}}^n(u)\right\|_2^2\right] + 2\mathbb{E}_{u\sim\gamma}\left[\left\|\Gamma_d^n \circ E_{\mathcal{X}}^n(u) - E_{\mathcal{Y}}^n \circ \Psi(u)\right\|_2^2\right] \\
&\leq 2d_{\mathcal{Y}}\varepsilon_1^2 + 2\mathbb{E}_{u\sim\gamma}\left[\left\|\Gamma_d^n \circ E_{\mathcal{X}}^n(u) - E_{\mathcal{Y}}^n \circ \Psi(u)\right\|_2^2\right] \\
&= 2d_{\mathcal{Y}}\varepsilon_1^2 + 2\mathbb{E}_{u\sim\gamma}\left[\left\|E_{\mathcal{Y}}^n \circ \Psi \circ D_{\mathcal{X}}^n \circ E_{\mathcal{X}}^n(u) - E_{\mathcal{Y}}^n \circ \Psi(u)\right\|_2^2\right] \quad \text{by the definition of } \Gamma_d \text{ in (52)} \\
&\leq 2d_{\mathcal{Y}}\varepsilon_1^2 + 2L_{E_{\mathcal{Y}}^n}^2 L_\Psi^2 \mathbb{E}_{u\sim\gamma}\left[\|D_{\mathcal{X}}^n \circ E_{\mathcal{X}}^n(u) - u\|_{\mathcal{X}}^2\right] \\
&= 2d_{\mathcal{Y}}\varepsilon_1^2 + 2L_{E_{\mathcal{Y}}^n}^2 L_\Psi^2 \mathbb{E}_{u\sim\gamma}\left[\left\|\Pi_{\mathcal{X},d_{\mathcal{X}}}^n(u) - u\right\|_{\mathcal{X}}^2\right]. \qquad (56)
\end{aligned}
$$

An upper bound of the second term in (54) is provided by the following lemma (see a proof in Appendix D):

Lemma 24 *Under the conditions of Theorem 3, for any $\delta \in (0,1)$, we have*

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\langle \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u_i), \boldsymbol{\epsilon}_i\rangle\right] \\
&\leq 2\sqrt{2d_{\mathcal{Y}}}\sigma\left(\sqrt{\mathbb{E}_{\mathcal{S}_2}\|\Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u_i) - \Gamma_d^n \circ E_{\mathcal{X}}^n(u_i)\|_n^2} + \sqrt{d_{\mathcal{Y}}}\delta\right)\sqrt{\frac{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, \|\cdot\|_\infty) + 2}{n}} + d_{\mathcal{Y}}\sigma\delta. \qquad (57)
\end{aligned}
$$

Let $\mathcal{F}_{\mathrm{NN}}$ be the network architecture specified in (55). Substituting (56) and (57) into (54), we have

$$
\begin{aligned}
\mathrm{T}_1 =& 2\mathbb{E}_{\mathcal{S}_2}\left[\left\|\Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n \circ \Psi(u_i)\right\|_n^2\right] \\
\leq& 4d_{\mathcal{Y}}\varepsilon_1^2 + 8\sqrt{2d_{\mathcal{Y}}}\sigma\left(\sqrt{\mathbb{E}_{\mathcal{S}_2}\left\|\Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u_i) - \Gamma_d^n \circ E_{\mathcal{X}}^n(u_i)\right\|_n^2} + \sqrt{d_{\mathcal{Y}}}\delta\right)\sqrt{\frac{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, \|\cdot\|_\infty) + 2}{n}} \\
&+ 4d_{\mathcal{Y}}\sigma\delta + 4L_{E_{\mathcal{Y}}^n}^2 L_\Psi^2 \mathbb{E}_{u\sim\gamma}\left[\left\|\Pi_{\mathcal{X},d_{\mathcal{X}}}^n(u) - u\right\|_{\mathcal{X}}^2\right]. \qquad (58)
\end{aligned}
$$

Denote

$$
\begin{aligned}
&\rho = \sqrt{\mathbb{E}_{\mathcal{S}_2}\left[\left\|\Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u_i) - E_{\mathcal{Y}}^n \circ \Psi(u_i)\right\|_n^2\right]}, \\
&a = 2d_{\mathcal{Y}}\varepsilon_1^2 + 2d_{\mathcal{Y}}\sigma\delta + 2L_{E_{\mathcal{Y}}^n}^2 L_\Psi^2 \mathbb{E}_{u\sim\gamma}\left[\left\|\Pi_{\mathcal{X},d_{\mathcal{X}}}^n(u) - u\right\|_{\mathcal{X}}^2\right] + 4\sqrt{2}d_{\mathcal{Y}}\sigma\delta\sqrt{\frac{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, \|\cdot\|_\infty) + 2}{n}}, \\
&b = 2\sqrt{2d_{\mathcal{Y}}}\sigma\sqrt{\frac{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, \|\cdot\|_\infty) + 2}{n}}.
\end{aligned}
$$

Inequality (58) can be rewritten as

$$
\rho^2 \leq a + 2b\rho,
$$

from which we deduce that

$$(\rho - b)^2 \le a + b^2 \Rightarrow \rho^2 \le 2a + 4b^2.$$

Therefore,

$$\begin{aligned}\mathrm{T}_1 =& 2\rho^2 \le 8d_{\mathcal{Y}}\varepsilon_1^2 + 64d_{\mathcal{Y}}\sigma^2\frac{\log\mathcal{N}(\delta,\mathcal{F}_{\mathrm{NN}},\|\cdot\|_\infty)+2}{n} + 16\sqrt{2}d_{\mathcal{Y}}\sigma\delta\sqrt{\frac{\log\mathcal{N}(\delta,\mathcal{F}_{\mathrm{NN}},\|\cdot\|_\infty)+2}{n}}\\ &+ 8d_{\mathcal{Y}}\sigma\delta + 8L_{E_{\mathcal{Y}}^n}^2L_\Psi^2\mathbb{E}_{u\sim\gamma}\left[\left\|\Pi_{\mathcal{X},d_{\mathcal{X}}}^n(u)-u\right\|_{\mathcal{X}}^2\right]. \end{aligned} \tag{59}$$

Upper bound of T_2. The term T_2 is the difference between the population risk and the empirical risk of the network estimator Γ_{NN}, while there is a factor 2 ahead of the empirical risk. Utilizing a covering of $\mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}},L,p,K,\kappa,M)$ and Bernstein-type inequalities, we establish a fast convergence of T_2. The upper bound is presented in the following lemma (see a proof in Appendix E).

Lemma 25 *Under the conditions of Theorem 3, we have*

$$\mathrm{T}_2 \le \frac{35d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{E_{\mathcal{Y}}}^2}{n}\log\mathcal{N}\left(\frac{\delta}{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}R_{\mathcal{Y}}},\mathcal{F}_{\mathrm{NN}},\|\cdot\|_\infty\right)+6\delta. \tag{60}$$

Substituting (59) and (60) into (50) gives rise to

$$\begin{aligned}\mathrm{I} \le& 2L_{D_{\mathcal{Y}}^n}^2\mathbb{E}_{\mathcal{S}_1}\mathbb{E}_{u\sim\gamma}\left[\left\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u)-E_{\mathcal{Y}}^n\circ\Psi(u)\right\|_2^2\right]\\ =& 2L_{D_{\mathcal{Y}}^n}^2\mathbb{E}_{\mathcal{S}_1}[\mathrm{T}_1]+2L_{D_{\mathcal{Y}}^n}^2\mathbb{E}_{\mathcal{S}_1}[\mathrm{T}_2]\\ \le& 16d_{\mathcal{Y}}L_{D_{\mathcal{Y}}^n}^2\varepsilon_1^2+128d_{\mathcal{Y}}\sigma^2L_{D_{\mathcal{Y}}^n}^2\frac{\log\mathcal{N}(\delta,\mathcal{F}_{\mathrm{NN}},\|\cdot\|_\infty)+2}{n}\\ &+32\sqrt{2}d_{\mathcal{Y}}\sigma L_{D_{\mathcal{Y}}^n}^2\delta\sqrt{\frac{\log\mathcal{N}(\delta,\mathcal{F}_{\mathrm{NN}},\|\cdot\|_\infty)+2}{n}}\\ &+16d_{\mathcal{Y}}\sigma L_{D_{\mathcal{Y}}^n}^2\delta+16L_{D_{\mathcal{Y}}^n}^2L_{E_{\mathcal{Y}}^n}^2L_\Psi^2\mathbb{E}_{u\sim\gamma}\|\Pi_{\mathcal{X},d_{\mathcal{X}}}^n(u)-u\|_{\mathcal{X}}^2\\ &+\frac{70d_{\mathcal{Y}}L_{D_{\mathcal{Y}}^n}^2L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}{n}\log\mathcal{N}\left(\frac{\delta}{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}R_{\mathcal{Y}}},\mathcal{F}_{\mathrm{NN}},\|\cdot\|_\infty\right)+12L_{D_{\mathcal{Y}}^n}^2\delta\\ \le& 16d_{\mathcal{Y}}L_{D_{\mathcal{Y}}^n}^2\varepsilon_1^2+\frac{128d_{\mathcal{Y}}\sigma^2L_{D_{\mathcal{Y}}^n}^2+70d_{\mathcal{Y}}L_{D_{\mathcal{Y}}^n}^2L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}{n}\log\mathcal{N}\left(\frac{\delta}{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}R_{\mathcal{Y}}},\mathcal{F}_{\mathrm{NN}},\|\cdot\|_\infty\right)\\ &+64d_{\mathcal{Y}}\sigma L_{D_{\mathcal{Y}}^n}^2\delta\sqrt{\frac{\log\mathcal{N}(\delta,\mathcal{F}_{\mathrm{NN}},\|\cdot\|_\infty)}{n}}+(16d_{\mathcal{Y}}\sigma+12)L_{D_{\mathcal{Y}}^n}^2\delta\\ &+16L_{D_{\mathcal{Y}}^n}^2L_{E_{\mathcal{Y}}^n}^2L_\Psi^2\mathbb{E}_{u\sim\gamma}\|\Pi_{\mathcal{X},d_{\mathcal{X}}}^n(u)-u\|_{\mathcal{X}}^2, \end{aligned} \tag{61}$$

when $\delta<1$. The covering number of $\mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}},L,p,K,\kappa,M)$ can be bounded in terms of its parameters, which is summarized in the following lemma:

Lemma 26 (Lemma 6 of Chen et al. (2022)) *Let $\mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}}, L, p, K, \kappa, M)$ be a class of network: $[-B, B]^{d_{\mathcal{X}}} \to [-M, M]^{d_{\mathcal{Y}}}$. For any $\delta > 0$, the δ-covering number of $\mathcal{F}_{\mathrm{NN}}(L, p, K, \kappa, M)$ is bounded by*

$$\mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}}, L, p, K, \kappa, M), \|\cdot\|_\infty) \leq \left(\frac{2L^2(pB+2)\kappa^L p^{L+1}}{\delta}\right)^{d_{\mathcal{Y}}K}. \tag{62}$$

Combining (55) and (62) gives

$$\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}}, L, p, K, \kappa, M), \|\cdot\|_\infty) \leq C_7 d_{\mathcal{Y}} \left(\varepsilon_1^{-d_{\mathcal{X}}} \log^3 \varepsilon_1^{-1} + \log\delta + \log d_{\mathcal{Y}}\right), \tag{63}$$

where C_7 is a constant depending on $d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{E_{\mathcal{X}}^n}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}^n}$ and L_Ψ. Substituting (63) into (61) yields

$$\begin{aligned}
\mathrm{I} \leq & 16 d_{\mathcal{Y}} L_{D_{\mathcal{Y}}^n}^2 \varepsilon_1^2 + C_7 d_{\mathcal{Y}}^2 L_{D_{\mathcal{Y}}^n}^2 \frac{128\sigma^2 + 70 L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2}{n} \left(\varepsilon_1^{-d_{\mathcal{X}}} \log^3 \varepsilon_1^{-1} + \log\delta + \log d_{\mathcal{Y}}\right) \\
& + 64 d_{\mathcal{Y}} \sigma L_{D_{\mathcal{Y}}^n}^2 \delta \sqrt{\frac{C_7 d_{\mathcal{Y}} \left(\varepsilon_1^{-d_{\mathcal{X}}} \log^3 \varepsilon_1^{-1} + \log\delta + \log d_{\mathcal{Y}}\right)}{n}} \\
& + (16 d_{\mathcal{Y}}\sigma + 12) L_{D_{\mathcal{Y}}^n}^2 \delta + 16 L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_\Psi^2 \mathbb{E}_{u\sim\gamma}\left[\left\|\Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u) - u\right\|_{\mathcal{X}}^2\right].
\end{aligned} \tag{64}$$

Setting

$$\varepsilon_1 = d_{\mathcal{Y}}^{\frac{1}{2+d_{\mathcal{X}}}} n^{-\frac{1}{2+d_{\mathcal{X}}}}, \delta = n^{-1},$$

we get an upper bound of I

$$\begin{aligned}
\mathrm{I} \leq & C_1(\sigma^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^3 n + C_2(\sigma^2 + R_{\mathcal{Y}}^2) d_{\mathcal{Y}}^2 (\log d_{\mathcal{Y}}) n^{-1} \\
& + 16 L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_\Psi^2 \mathbb{E}_{\mathcal{S}_1} \mathbb{E}_{u\sim\gamma}\left[\left\|\Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u) - u\right\|_2^2\right]
\end{aligned} \tag{65}$$

for some constants C_1, C_2 depending on $d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{E_{\mathcal{X}}^n}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}^n}, L_\Psi$. The constants C_1, C_2 are the same ones as in Theorem 3. The resulting network architecture $\mathcal{F}(d_{\mathcal{Y}}, L, p, K, \kappa, M)$ has

$$\begin{aligned}
& L = O(\log n + \log d_{\mathcal{Y}}),\ p = O\left(d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{2+d_{\mathcal{X}}}}\right),\ K = O\left(d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} \log n\right), \\
& \kappa = \max\left\{1, \sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}, \sqrt{d_{\mathcal{X}}} L_{E_{\mathcal{X}}^n} R_{\mathcal{X}}, \sqrt{d_{\mathcal{X}}} L_{E_{\mathcal{X}}^n} L_{E_{\mathcal{Y}}^n} L_{D_{\mathcal{X}}^n} L_\Psi R_{\mathcal{X}}\right\},\ M = \sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}.
\end{aligned} \tag{66}$$

Combining the bounds of I and II. Putting (64) and (49) together gives rise to

$$\begin{aligned}
& \mathbb{E}_{\mathcal{S}} \mathbb{E}_{u\sim\gamma}\left[\left\|D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - \Psi(u)\right\|_{\mathcal{Y}}^2\right] \\
\leq & \mathrm{I} + \mathrm{II}
\end{aligned}$$

$$\begin{aligned}&\leq C_1(\sigma^2+R_{\mathcal{Y}}^2)d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}}n^{-\frac{2}{2+d_{\mathcal{X}}}}\log^3 n + C_2(\sigma^2+R_{\mathcal{Y}}^2)d_{\mathcal{Y}}^2(\log d_{\mathcal{Y}})n^{-1}\\ &\quad + 16L_{D_{\mathcal{Y}}^n}^2L_{E_{\mathcal{Y}}^n}^2L_\Psi^2\mathbb{E}_{\mathcal{S}_1}\mathbb{E}_{u\sim\gamma}\left[\left\|\Pi_{\mathcal{X},d_{\mathcal{X}}}^n(u)-u\right\|_2^2\right] + 2\mathbb{E}_{\mathcal{S}_1}\mathbb{E}_{v^*\sim\Psi_{\#}\gamma}\left[\left\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}^n(v^*)-v^*\right\|_{\mathcal{Y}}^2\right],\end{aligned} \tag{67}$$

where C_1, C_2 are constants depending on $d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{E_{\mathcal{X}}^n}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}^n}, L_\Psi$. Substituting $\sigma = L_{E_{\mathcal{Y}}^n}\widetilde{\sigma}$, the theorem is proved. ■

7.3 Proof of Theorem 4

Proof of Theorem 4. The main framework of the proof of Theorem 4 is the same as that of Theorem 3, except special attentions need to be paid on bounding T_1 and T_2 in (51):

- For T_1, we establish a new result on the approximation error of deep neural networks with architecture $\mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}}, L, p, M)$.
- For T_2, we derive an upper bound using the uniform covering numbers. The motivation to use $\mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}}, L, p, M)$ is that it removes parameter upper bound, which is appealing to practical training. However, removing parameter upper bound leads to technical issues in bounding T_2. We address these issues using the uniform covering numbers thanks to the boundedness of network outputs inspired by Jiao et al. (2021).

The first part of our proof is the same as that of Theorem 3 up to (51), which is omitted here. In the following, we bound T_1 and T_2 in order.

Upper bound of T_1. The upper bound of T_1 can be derived similarly as that in Section 7.2, except we make two changes:

- Replace Lemma 22 by the following one

  Lemma 27 (Theorem 1.1 of Shen et al. (2020)) *Let $0<\alpha\leq 1$ be a real number. There exists a FNN architecture $\mathcal{F}_{\mathrm{NN}}(1, L, p, M)$ with $d_{\mathcal{Y}}=1$ such that for any integers $\widetilde{L}, \widetilde{p}>0$ and $f\in\mathcal{C}^{0,\alpha}([-B,B]^d)$ with $\|f\|_{\mathcal{C}^{0,\alpha}}\leq R$, such an architecture gives rise to an FNN $\widetilde{f}$ with*

  $$\left\|\widetilde{f}-f\right\|_\infty \leq C\widetilde{L}^{-\frac{2\alpha}{d}}\widetilde{p}^{-\frac{2\alpha}{d}}$$

  for some constant C depending on α, d, B, R. This architecture has

  $$L=O(\widetilde{L}),\ p=O\left(\widetilde{p}\right),\ M=R.$$

  The constant hidden in $O(\cdot)$ depends on α, d, B, R.

  According to Lemma 27 with $\alpha=1$, for any $\varepsilon_1>0$, there is a network architecture $\mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}}, L, p, M)$, such that for any Γ_{NN}^n defined in (52), there exists a $\widetilde{\Gamma}_d^n\in\mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}}, L, p, M)$ with

  $$\left\|\widetilde{\Gamma}_d^n-\Gamma_d^n\right\|_\infty\leq\varepsilon_1.$$

Such a network architecture has

$$L = O(\widetilde{L}),\ p = O\left(\widetilde{p}\right), M = \sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}, \tag{68}$$

where $\widetilde{L}, \widetilde{p} > 0$ are integers satisfying $\widetilde{L}\widetilde{p} = \left\lceil \varepsilon_1^{-d_{\mathcal{X}}/2} \right\rceil$. The constant hidden in $O(\cdot)$ depends on $d_{\mathcal{X}}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}^n}, L_\Psi, B$ and M.

- Replace Lemma 24 by

Lemma 28 *Under the conditions of Theorem 4, for any $\delta \in (0,1)$, we have*

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\langle \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u_i), \boldsymbol{\epsilon}_i\rangle\right] \\
\leq & 2\sqrt{2d_{\mathcal{Y}}}\sigma\left(\sqrt{\mathbb{E}_{\mathcal{S}_2}\|\Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u_i) - \Gamma_d^n \circ E_{\mathcal{X}}^n(u_i)\|_n^2} + \sqrt{d_{\mathcal{Y}}}\delta\right)\sqrt{\frac{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, n) + 2}{n}} + d_{\mathcal{Y}}\sigma\delta.
\end{aligned} \tag{69}$$

Lemma 28 can be proved similarly as Lemma 24. We need to replace the δ-cover $\mathcal{F}^* = \{\Gamma_j^*\}_{j=1}^{\mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, \|\cdot\|_\infty)}$ by a δ-cover of $\mathcal{F}_{\mathrm{NN}}$ with respect to $\mathcal{S}_2$: $\mathcal{F}^* = \{\Gamma_j^*\}_{j=1}^{\mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, n)}$, where $\mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, n)$ is the uniform covering number. Here the cover $\mathcal{F}^*$ depends on the samples $\{E_{\mathcal{X}}^n(u_i)\}_{i=n+1}^{2n}$. Then there exists $\Gamma^* \in \mathcal{F}^*$ satisfying $\|\Gamma^* \circ E_{\mathcal{X}}^n(u_i) - \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u_i)\|_\infty \leq \delta$ for any $n+1 \leq i \leq 2n$. The proof is omitted here.

Following the rest of the proof for T_1 in Section 7.2, we can derive that

$$\begin{aligned}
\mathrm{T}_1 \leq & 8d_{\mathcal{Y}}\varepsilon_1^2 + 64d_{\mathcal{Y}}\sigma^2\frac{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, n) + 2}{n} + 16\sqrt{2}d_{\mathcal{Y}}\sigma\delta\sqrt{\frac{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, n) + 2}{n}} \\
& + 8d_{\mathcal{Y}}\sigma\delta + 8L_{E_{\mathcal{Y}}^n}^2 L_\Psi^2 \mathbb{E}_{u\sim\gamma}\left[\left\|\Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u) - u\right\|_{\mathcal{X}}^2\right].
\end{aligned} \tag{70}$$

The network architecture of $\mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}}, L, p, M)$ is specified in (68).

Upper bound of T_2. Using the covering number defined in Definition 20, we have the following bound of T_2.

Lemma 29 *Under the conditions of Theorem 4, we have*

$$\mathrm{T}_2 \leq \frac{35d_{\mathcal{Y}}R_{\mathcal{Y}}^2}{n}\log \mathcal{N}\left(\frac{\delta}{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}R_{\mathcal{Y}}}, \mathcal{F}_{\mathrm{NN}}, 2n\right) + 6\delta. \tag{71}$$

Lemma 29 is proved in Appendix F using techniques similar to those in the proof of Lemma 25. Substituting (70) and (71) into (50) gives rise to

$$\begin{aligned}
\mathrm{I} \leq & 2L_{D_{\mathcal{Y}}}^2 \mathbb{E}_{\mathcal{S}_1}\mathbb{E}_{u\sim\gamma}\left[\left\|\Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u) - E_{\mathcal{Y}}^n \circ \Psi(u)\right\|_2^2\right] \\
= & 2L_{D_{\mathcal{Y}}^n}^2 \mathbb{E}_{\mathcal{S}_1}[\mathrm{T}_1] + 2L_{D_{\mathcal{Y}}^n}^2 \mathbb{E}_{\mathcal{S}_1}[\mathrm{T}_2]
\end{aligned}$$

$$
\begin{aligned}
\leq & 16 d_{\mathcal{Y}} L_{D_{\mathcal{Y}}^n}^2 \varepsilon_1^2 + 128 d_{\mathcal{Y}} \sigma^2 L_{D_{\mathcal{Y}}^n}^2 \frac{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, n) + 2}{n} + 32\sqrt{2} d_{\mathcal{Y}} \sigma L_{D_{\mathcal{Y}}^n}^2 \delta \sqrt{\frac{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, n) + 2}{n}} \\
& + 16 d_{\mathcal{Y}} \sigma L_{D_{\mathcal{Y}}^n}^2 \delta + 16 L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_{\Psi}^2 \mathbb{E}_{u \sim \gamma} \left[\left\| \Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u) - u \right\|_{\mathcal{X}}^2 \right] \qquad (72) \\
& + \frac{70 d_{\mathcal{Y}} L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2}{n} \log \mathcal{N} \left(\frac{\delta}{4 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}}, \mathcal{F}_{\mathrm{NN}}, 2n \right) + 12 L_{D_{\mathcal{Y}}^n}^2 \delta \\
\leq & 16 d_{\mathcal{Y}} L_{D_{\mathcal{Y}}^n}^2 \varepsilon_1^2 + \frac{128 d_{\mathcal{Y}} \sigma^2 L_{D_{\mathcal{Y}}^n}^2 + 70 d_{\mathcal{Y}} L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2}{3n} \log \mathcal{N} \left(\frac{\delta}{4 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}}, \mathcal{F}_{\mathrm{NN}}, 2n \right) \\
& + 64 d_{\mathcal{Y}} \sigma L_{D_{\mathcal{Y}}^n}^2 \delta \sqrt{\frac{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, n)}{n}} \\
& + (16 d_{\mathcal{Y}} \sigma + 12) L_{D_{\mathcal{Y}}^n}^2 \delta + 16 L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_{\Psi}^2 \mathbb{E}_{u \sim \gamma} \left[\left\| \Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u) - u \right\|_{\mathcal{X}}^2 \right]. \qquad (73)
\end{aligned}
$$

The covering number in (73) can be bounded using the pseudo-dimension of the network class:

Lemma 30 (Theorem 12.2 of Anthony and Bartlett (1999)) *Let F be a class of functions from some domain Ω to $[-M, M]$. Denote the pseudo-dimension of F by $\mathrm{Pdim}(F)$. For any $\delta > 0$, we have*

$$
\mathcal{N}(\delta, F, m) \leq \left(\frac{2eMm}{\delta \mathrm{Pdim}(F)} \right)^{\mathrm{Pdim}(F)}
$$

for $m > \mathrm{Pdim}(F)$.

The next lemma shows that the pseudo-dimension of $\mathcal{F}_{\mathrm{NN}}(1, L, p, M)$ can be bounded using its parameters:

Lemma 31 (Theorem 7 of Bartlett et al. (2019)) *For any network architecture $\mathcal{F}_{\mathrm{NN}}$ with L layers and U parameters, there exists a universal constant C such that*

$$
\mathrm{Pdim}(\mathcal{F}_{\mathrm{NN}}) \leq CLU \log(U).
$$

Now conciser the network architecture $\mathcal{F}_{\mathrm{NN}}(1, L, p, M)$, the number of parameters is bounded by $U = Lp^2$. Combing Lemma 30 and 31, we have

$$
\log \mathcal{N} \left(\frac{\delta}{4 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}}, \mathcal{F}_{\mathrm{NN}}(d_{\mathcal{Y}}, L, p, M), 2n \right) \leq C_8 d_{\mathcal{Y}} p^2 L^2 \log \left(p^2 L \right) \left(\log M + \log \delta^{-1} + \log n \right) \qquad (74)
$$

when $2n > C_9 p^2 L^2 \log(p^2 L)$ for some universal constant C_8, C_9 . Substituting (68) into (74) gives rise to

$$
\log \mathcal{N} \left(\frac{\delta}{4 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}}, \mathcal{F}_{\mathrm{NN}}, 2n \right) \leq C_8 d_{\mathcal{Y}} \varepsilon_1^{-d_{\mathcal{X}}} \log \left(\varepsilon_1^{-1} \right) \left(\log \delta^{-1} + \log n \right). \qquad (75)
$$

Substituting (75) into (73) yields

$$\begin{aligned}\mathrm{I} \leq & 16 d_{\mathcal{Y}} L_{D_{\mathcal{Y}}^n}^2 \varepsilon_1^2+C_8 d_{\mathcal{Y}}^2 L_{D_{\mathcal{Y}}^n}^2 \frac{128 \sigma^2+70 L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2}{n} \varepsilon_1^{-d_{\mathcal{X}}} \log \left(\varepsilon_1^{-1}\right)\left(\log \delta^{-1}+\log n\right) \\ & +64 d_{\mathcal{Y}} \sigma L_{D_{\mathcal{Y}}^n}^2 \delta \sqrt{\frac{C_8 d_{\mathcal{Y}} \varepsilon_1^{-d_{\mathcal{X}}} \log \left(\varepsilon_1^{-1}\right)\left(\log \delta^{-1}+\log n\right)}{n}} \\ & +\left(16 d_{\mathcal{Y}} \sigma+12\right) L_{D_{\mathcal{Y}}^n}^2 \delta+16 L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_{\Psi}^2 \mathbb{E}_{u \sim \gamma}\left[\left\|\Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u)-u\right\|_{\mathcal{X}}^2\right] . \end{aligned} \tag{76}$$

Setting

$$\varepsilon_1=d_{\mathcal{Y}}^{\frac{1}{2+d_{\mathcal{X}}}} n^{-\frac{1}{2+d_{\mathcal{X}}}}, \delta=n^{-1},$$

we have

$$\mathrm{I} \leq C_4\left(\sigma^2+R_{\mathcal{Y}}^2\right) d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log ^2 n+16 L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_{\Psi}^2 \mathbb{E}_{u \sim \gamma}\left[\left\|\Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u)-u\right\|_{\mathcal{X}}^2\right], \tag{77}$$

where C_4 is a constant depending on $d_{\mathcal{X}}, R_{\mathcal{X}}, R_{\mathcal{Y}}, L_{E_{\mathcal{X}}^n}, L_{E_{\mathcal{Y}}^n}, L_{D_{\mathcal{X}}^n}, L_{\Psi}$, the same constant in Theorem 4. The resulting network architecture $\mathcal{F}(L, p, M)$ has

$$L=O(\widetilde{L}),\ p=O\left(\widetilde{p}\right), M=\sqrt{d_{\mathcal{Y}}} L_{E_{\mathcal{Y}}^n} R_{\mathcal{Y}}, \tag{78}$$

where $\widetilde{L} \widetilde{p}=d_{\mathcal{Y}}^{-\frac{d_{\mathcal{X}}}{4+2 d_{\mathcal{X}}}} n^{\frac{d_{\mathcal{X}}}{4+2 d_{\mathcal{X}}}}$. Now we check the condition in Lemma 30. Under the choice of L and p above, we have

$$L^2 p^2 \log \left(p^2 L\right)=O\left(n^{\frac{2 d_{\mathcal{X}}}{4+2 d_{\mathcal{X}}}} \log n\right)<2 n$$

when n is large enough. The condition is satisfied.

Combining the bounds of I and II. Putting (77) and (49) together gives rise to

$$\begin{aligned} & \mathbb{E}_{\mathcal{S}} \mathbb{E}_{u \sim \gamma}\left[\left\|D_{\mathcal{Y}}^n \circ \Gamma_{\mathrm{NN}} \circ E_{\mathcal{X}}^n(u)-\Psi(u)\right\|_{\mathcal{Y}}^2\right] \\ \leq & \mathrm{I}+\mathrm{II} \\ \leq & C_4\left(\sigma^2+R_{\mathcal{Y}}^2\right) d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log ^2 n \\ & +16 L_{D_{\mathcal{Y}}^n}^2 L_{E_{\mathcal{Y}}^n}^2 L_{\Psi}^2 \mathbb{E}_{u \sim \gamma}\left[\left\|\Pi_{\mathcal{X}, d_{\mathcal{X}}}^n(u)-u\right\|_{\mathcal{X}}^2\right]+2 \mathbb{E}_{\mathcal{S}_1} \mathbb{E}_{v^* \sim \Psi_{\#} \gamma}\left[\left\|\Pi_{\mathcal{Y}, d_{\mathcal{Y}}}^n\left(v^*\right)-v^*\right\|_{\mathcal{Y}}^2\right] . \end{aligned} \tag{79}$$

Substituting $\sigma=L_{E_{\mathcal{Y}}^n} \widetilde{\sigma}$ finishes the proof. ■

7.4 Proof of Corollary 10

Proof of Corollary 10. We only need to derive upper bounds of

$$\mathbb{E}_{u \sim \gamma}\left[\left\|\Pi_{\mathcal{X}, d_{\mathcal{X}}}(u)-u\right\|_2^2\right] \text{ and } \mathbb{E}_{v \sim \Psi_{\#} \gamma}\left[\left\|\Pi_{\mathcal{Y}, d_{\mathcal{Y}}}(v)-v\right\|_{\mathcal{Y}}^2\right].$$

Then Corollary 10 is a direct result of Corollary 8. Our proof relies on the following lemma which gives an approximation error of Legendre polynomials for Hölder functions:

Lemma 32 (Theorem 4.5(ii) of Schultz (1969)) *Let $k \geq 0$ be an integer and $\alpha > 0$. For any $f \in \mathcal{C}^{k,\alpha}([0,1]^D)$ with $\|f\|_{\mathcal{C}^{k,\alpha}} < \infty$, there exists $\widetilde{f} \in \text{span}(\Phi^{\text{L},r})$ such that*

$$\left\|f - \widetilde{f}\right\|_\infty \leq \frac{C}{r^{k+\alpha}},$$

where C is a constant depending on D and $\|f\|_{\mathcal{C}^{k,\alpha}}$.

We first derive an upper bound of $\mathbb{E}_{u\sim\gamma}\left[\|\Pi_{\mathcal{X},d_\mathcal{X}}(u) - u\|_2^2\right]$. For any $u \in \Omega_\mathcal{X}$, according to Lemma 32, there exists $\widetilde{u} \in \text{span}(\Phi^{\text{L},r_\mathcal{X}})$ such that

$$\|u - \widetilde{u}\|_\infty \leq C_{10} r_\mathcal{X}^{-s},$$

where $s = k + \alpha$, C_{10} is a constant depending on D and $C_{\mathcal{H}_P,\mathcal{X}}$. We deduce that

$$\begin{aligned}
\|\Pi_{\mathcal{X},d_\mathcal{X}}(u) - u\|_\mathcal{X}^2 =& \min_{\bar{u}\in\text{span}(\Phi^{\text{L},r_\mathcal{X}})} \|\bar{u} - u\|_\mathcal{X}^2 \\
\leq& \|\widetilde{u} - u\|_\mathcal{X}^2 \\
\leq& \int_{[-1,1]^D} |\widetilde{u} - u|^2 d\boldsymbol{x} \\
\leq& 2^D C_{10} r_\mathcal{X}^{-2s} \\
=& 2^D C_{10} d_\mathcal{X}^{-\frac{2s}{D}},
\end{aligned}$$

where in the last equality $d_\mathcal{X} = r_\mathcal{X}^D$ is used. Therefore

$$\mathbb{E}_{u\sim\gamma}\left[\|\Pi_{\mathcal{X},d_\mathcal{X}}(u) - u\|_2^2\right] \leq C_5 d_\mathcal{X}^{-\frac{2s}{D}},$$

where C_5 is a constant depending on D and $C_{\mathcal{H}_P,\mathcal{X}}$. Similarly, one can show

$$\mathbb{E}_{v\sim\Psi_{\#}\gamma}\left[\left\|\Pi_{\mathcal{Y},d_\mathcal{Y}}(v) - v\right\|_\mathcal{Y}^2\right] \leq C_6 d_\mathcal{Y}^{-\frac{2s}{D}},$$

where C_6 is a constant depending on D and $C_{\mathcal{H},\mathcal{Y}}$. The theorem is proved. ■

7.5 Proof of Corollary 11

Proof of Corollary 11. Our proof relies on the following lemma which gives an approximation error of trigonometric bases for periodic Hölder functions.

Lemma 33 (Theorem 4.3(ii) of Schultz (1969)) *Let $k \geq 0$ be an integer and $0 < \alpha \leq 1$. For any $f \in \mathcal{P} \cap \mathcal{C}^{k,\alpha}\left([0,1]^D\right)$ with $\|f\|_{\mathcal{C}^{k,\alpha}} < \infty$, there exists $\widetilde{f} \in \text{span}(\Phi^{\text{T},r})$ such that*

$$\left\|f - \widetilde{f}\right\|_\infty \leq \frac{C}{r^s},$$

where C is a constant depending on D and $\|f\|_{\mathcal{C}^{k,\alpha}}$.

Corollary 11 can be proved by following the proof of Corollary 11 in which Lemma 32 is replaced by Lemma 33. ■

7.6 Proof of Theorem 13

Proof of Theorem 13. Lemma 12 implies that $E^n_{\mathcal{X}}, D^n_{\mathcal{X}}, E^n_{\mathcal{Y}}, D^n_{\mathcal{Y}}$ are Lipschitz with a Lipschitz constant 1. Therefore Corollary 8 can be applied. We only need to bound $\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\left\|\Pi^n_{\mathcal{X},d_{\mathcal{X}}}(u)-u\right\|^2_{\mathcal{X}}\right]$ and $\mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi^n_{\mathcal{Y},d_{\mathcal{Y}}}(w)-w\right\|^2_{\mathcal{Y}}\right]$ in (21). We use the following lemma:

Lemma 34 (Theorem 3.4 of Bhattacharya et al. (2021)) *Let $\mathcal{H}$ be a separable Hilbert space and ρ be a probabillity measure defined on it. Define the covariance operator $G_\rho = \mathbb{E}_{u\sim\rho}u\otimes u$ and its empirical estimation from n samples by $G^n_\rho = \frac{1}{n}\sum_{i=1}^n u_i\otimes u_i$ where $\{u_i\}_{i=1}^n$ are i.i.d. samples sampled from ρ. For some integer $d>0$, let $\Pi_{\mathcal{H},d}$ and $\Pi^n_{\mathcal{H},d}$ be the projectors that project any $u\in\mathcal{H}$ to the space spanned by the eigenfunctions corresponding to the largest d eigenvalues of G_ρ and G^n_ρ, respectively. We have*

$$\mathbb{E}_{\{u_k\}_{k=1}^n\sim\rho}\mathbb{E}_{u\sim\rho}\left[\left\|\Pi^n_{\mathcal{H},d}(u)-u\right\|^2_{\mathcal{H}}\right]\le\sqrt{\frac{Cd}{n}}+\mathbb{E}_{u\sim\rho}\left[\left\|\Pi_{\mathcal{H},d}(u)-u\right\|^2_{\mathcal{H}}\right]$$

with $C=\mathbb{E}_{\{u_i\}_{i=1}^n\sim\rho}\left[\|G^n-G\|^2_{\mathrm{HS}}\right]$, where $\|\cdot\|_{\mathrm{HS}}$ is the Hilbert-Schmidt norm.

We first bound $\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\left\|\Pi^n_{\mathcal{X},d_{\mathcal{X}}}(u)-u\right\|^2_{\mathcal{X}}\right]$. For any $u\sim\gamma$, we have $\|u\|_{\mathcal{X}}\le R_{\mathcal{X}}$. Therefore

$$\mathbb{E}_{u\sim\gamma}\left[\|G^n_{\mathcal{X}}-G_{\mathcal{X}}\|^2_{\mathrm{HS}}\right]\le 4\mathbb{E}_{u\sim\gamma}\left[\|u\|^4_{\mathcal{X}}\right]\le 4R^4_{\mathcal{X}}$$

and Lemma 34 gives

$$\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\left\|\Pi^n_{\mathcal{X},d_{\mathcal{X}}}(u)-u\right\|^2_{\mathcal{X}}\right]\le\sqrt{\frac{4R^4_{\mathcal{X}}d_{\mathcal{X}}}{n}}+\mathbb{E}_{u\sim\gamma}\left[\left\|\Pi_{\mathcal{X},d_{\mathcal{X}}}(u)-u\right\|^2_{\mathcal{X}}\right]. \tag{80}$$

An upper bound of $\mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi^n_{\mathcal{Y},d_{\mathcal{Y}}}(w)-w\right\|^2_{\mathcal{Y}}\right]$ is given by the following lemma (see a proof in Appendix G):

Lemma 35 *Under the conditions of Theorem 13, we have*

$$\begin{aligned}\mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi^n_{\mathcal{Y},d_{\mathcal{Y}}}(w)-w\right\|^2_{\mathcal{Y}}\right]\le&4\sqrt{\frac{(R_{\mathcal{Y}}+\widetilde{\sigma})^4d_{\mathcal{Y}}}{n}}+8\left(\frac{\widetilde{\sigma}}{\lambda_{d_{\mathcal{Y}}}-\lambda_{d_{\mathcal{Y}+1}}}\right)^2\widetilde{\sigma}^2(R_{\mathcal{Y}}+\widetilde{\sigma})^2\\&+10\widetilde{\sigma}^2+8\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}(w)-w\right\|^2_{\mathcal{Y}}\right].\end{aligned} \tag{81}$$

■

7.7 Proof of Corollary 14

Proof of Corollary 14. We only need to show that the eigenspace spanned by the first $d_{\mathcal{Y}}$ principal eigenfunctions of $G_{\Psi_{\#}\gamma}$ is the same as that of G_ζ. Then Corollary

14 can be proved by following the proof of Theorem 13 in which the upper bound of $\mathbb{E}_{\mathcal{S}}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi^n_{\mathcal{Y},d_{\mathcal{Y}}}(w)-w\right\|^2_{\mathcal{Y}}\right]$ can be derived in the same manner as that of $\mathbb{E}_{\mathcal{S}}\mathbb{E}_{u\sim\gamma}\left[\left\|\Pi^n_{\mathcal{X},d_{\mathcal{X}}}(u)-u\right\|^2_{\mathcal{X}}\right]$.

Denote the eigenvalues of $G_{\Psi_{\#}\gamma}$ in non-increasing order by $\{\lambda_{\Psi_{\#}\gamma,k}\}_{k=1}^{\infty}$. Denote the eigenspace spanned by the first $d_{\mathcal{Y}}$ principal eigenfunctions of $G_{\Psi_{\#}\gamma}$ by $\mathcal{K}$, and its compliment by $\mathcal{K}^{\top}$. Similarly, we define $\mathcal{K}_{\zeta}$ and $\mathcal{K}_{\zeta}^{\top}$ for G_{ζ}. From our assumption, $\mathcal{K}$ is also the eigenspace spanned by the first $d_{\mathcal{Y}}$ eigenfunctions of G_{μ}. We denote the eigenvalues of G_{μ} in non-increasing order by $\{\lambda_{\mu,k}\}_{k=1}^{\infty}$. We are going to show that $\mathcal{K}=\mathcal{K}_{\zeta}$. From (100), we have $G_{\zeta}=G_{\Psi_{\#}\gamma}+G_{\mu}$. Note that for any $\phi\in\mathcal{K}$ and $\widetilde{\phi}\in\mathcal{K}^{\top}$ with unit length, we have

$$\begin{aligned}\langle G_{\zeta}\phi,\phi\rangle_{\mathcal{Y}} =&\langle G_{\Psi_{\#}\gamma}\phi,\phi\rangle_{\mathcal{Y}}+\langle G_{\mu}\phi,\phi\rangle_{\mathcal{Y}}\\ \geq&\lambda_{\Psi_{\#},d_{\mathcal{Y}}}+\lambda_{\mu,d_{\mathcal{Y}}+1}\\ \geq&\lambda_{\Psi_{\#},d_{\mathcal{Y}}+1}+\lambda_{\mu,d_{\mathcal{Y}}+1}\\ \geq&\langle G_{\Psi_{\#}\gamma}\widetilde{\phi},\widetilde{\phi}\rangle_{\mathcal{Y}}+\langle G_{\mu}\widetilde{\phi},\widetilde{\phi}\rangle_{\mathcal{Y}}\\ =&\langle G_{\zeta}\widetilde{\phi},\widetilde{\phi}\rangle_{\mathcal{Y}}.\end{aligned}$$

Since both $\mathcal{K}$ and $\mathcal{K}_{\zeta}$ have dimension $d_{\mathcal{Y}}$, we have $\mathcal{K}=\mathcal{K}_{\zeta}$. The proof is finished. ■

7.8 Proof of Theorem 15

Proof of Theorem 15. Theorem 15 can be proved by following the proof of Theorem 4 with the following changes:

- Replace $E^n_{\mathcal{X}}$ by $E_{\mathcal{X}}$.
- Under Assumption 2 and 8, our target function $E_{\mathcal{Y}}\circ\Psi\circ D_{\mathcal{X}}$ is a Lipschitz function on $\mathcal{M}$. We replace Lemma 27 by the following one (see a proof in Appendix H):

  Lemma 36 *Suppose Assumption 8 holds. Assume for any $\mathbf{a}\in\mathcal{M}$, $\|\mathbf{a}\|_{\infty}\leq B$ for some $B>0$. There exists a FNN architecture $\mathcal{F}_{\mathrm{NN}}(1,L,p,M)$ such that for any integers $\widetilde{L},\widetilde{p}>0$ and $f\in\mathcal{C}^{0,1}(\mathcal{M})$ with $\|f\|_{\mathcal{C}^{0,1}}\leq R$, such an architecture gives rise to a FNN $\widetilde{f}$ with*

$$\left\|\widetilde{f}-f\right\|_{\infty}\leq C\widetilde{L}^{-\frac{2}{d_0}}\widetilde{p}^{-\frac{2}{d_0}}$$

  for some constant C depending on d_0,B,R,τ and the surface area of $\mathcal{M}$. This architecture has

$$L=O\left(\widetilde{L}\right),\ p=O\left(d_{\mathcal{X}}\widetilde{p}\right),\ M=R. \tag{82}$$

  The constant hidden in $O(\cdot)$ depends on d_0,B,R,τ and the surface area of $\mathcal{M}$.

■

7.9 Proof of Theorem 16

Proof of Theorem 16 Theorem 16 can be proved similarly as Theorem 15 while special attention needs to be paid on bounding $\log \mathcal{N}(\delta, \mathcal{F}_{\rm NN}(d_{\mathcal{Y}}, L, p, M), n)$. Note that the total number of parameters of $\mathcal{F}_{\rm NN}(d_{\mathcal{Y}}, L, p, M)$ is bounded by $U = Lp + d_{\mathcal{X}}p$. Combing Lemma 30 and 31, we have

$$\begin{aligned}&\log \mathcal{N}\left(\frac{\delta}{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}R_{\mathcal{Y}}}, \mathcal{F}_{\rm NN}(d_{\mathcal{Y}}, L, p, M), 2n\right)\\ \leq& C_{11}d_{\mathcal{Y}}(p^2L^2 + d_{\mathcal{X}}pL)\log\left(p^2L + d_{\mathcal{X}}Lp\right)\left(\log M + \log\delta^{-1} + \log n\right), \end{aligned} \tag{83}$$

where C_{11} is a universal constant. According to (68), one has $Lp = O\left(\varepsilon_1^{-d_{\mathcal{X}}/2}\log^2\left(\varepsilon^{-1}\right)\right)$. Using this relation and substituting the choice of L, p in (68) to (83) gives rise to

$$\begin{aligned}&\log \mathcal{N}\left(\frac{\delta}{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}R_{\mathcal{Y}}}, \mathcal{F}_{\rm NN}(d_{\mathcal{Y}}, L, p_1, p_2, M), 2n\right)\\ \leq& C_{11}d_{\mathcal{Y}}\left(\varepsilon_1^{-d_{\mathcal{X}}} + d_{\mathcal{X}}\varepsilon_1^{-d_{\mathcal{X}}/2}\right)\log\left(\varepsilon_1^{-1}\right)\left(\log\delta^{-1} + \log n\right). \end{aligned} \tag{84}$$

The proof can be finished by following the rest of the proof of Theorem 15. ■

8. Conclusion and discussion

We study the generalization error of a general framework on learning operators between infinite-dimensional spaces by two types of deep neural networks. With properly chosen encoders and decoders, our framework is discretization invariant. Our upper bound consists of a network estimation error and a projections error, and holds for general encoders and decoders under mild assumptions. The application of our results on some popular encoders and decoders are discussed, such as those using Legendre polynomials, trigonometric functions, and PCA. We also consider two scenarios where additional low dimensional structures of data can be exploited. The two scenarios are: (1) the input data can be encoded to vectors on a low dimensional manifold; (2) the operator has low complexity. In both scenarios, we show that the generalization error converges at a fast rate depending on the intrinsic dimension. Our results show that deep neural networks are adaptive to low dimensional structures of data in operator estimation. In general, our results provide a theoretical justification on the successes of deep neural networks for learning operators between infinite dimensional spaces.

As mentioned in Section 3.2, our network estimation error in Theorem 3 and 4 is optimal up to a logarithmic factor with respect to the sample size n for a fixed $d_{\mathcal{X}}$. While our bound has a factor $d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}}$, this term results from selecting the value of ε_1 by balancing the two terms $d_{\mathcal{Y}}\varepsilon_1^2$ and $d_{\mathcal{Y}}^2\varepsilon_1^{-d_{\mathcal{X}}}/n$, as in (64). It is an open question about whether our bound is minimax optimal with respect to $d_{\mathcal{Y}}$. In general, deriving a minimax rate for operator

learning by deep neural networks is intrinsically challenging due to its infinite-dimensional nature. In Lanthaler and Stuart (2023), the curse of dimensionality is revealed for learning functionals by deep neural networks. To achieve an ϵ approximation error of C^r functionals, the network size of PCANet, DeepONet, NOMAD and FNO is lower bounded in the order of $\exp(c\epsilon^{-1/(\alpha+1+\delta)r})$ (Lanthaler and Stuart, 2023, Theorem 2.15 and Proposition 2.22), with α, δ specified in Lanthaler and Stuart (2023). We will leave the investigation of minimax rates as our future work.

Acknowledgments

Hao Liu was partially supported by National Natural Science Foundation of China 12201530, HKRGC ECS 22302123 and HKBU 179356. Haizhao Yang was partially supported by the US National Science Foundation under awards DMS-2244988, DMS-2206333, and the Office of Naval Research Award N00014-23-1-2007. Tuo Zhao was partially supported by the US National Science Foundation under DMS-2012652. Wenjing Liao was partially supported by the US National Science Foundation under DMS-2012652, DMS-2145167 and U.S. Department of Energy under DE-SC0024348.

Appendix

Appendix A. The derivation for the error bound in Corollary 10 when $d_{\mathcal{X}} = d_{\mathcal{Y}} = \log^{\frac{1}{2}} n$

From Corollary 10, we need to balance the two terms $d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^6 n$ and $d_{\mathcal{X}}^{-\frac{2s}{D}}$. By setting $d_{\mathcal{X}} = d_{\mathcal{Y}} = \log^{\frac{1}{2}} n$, the first term decays faster than the second term as n increases. We want to find a lower bound of n, denoted by n_0, so that when $n > n_0$, the error is dominated by the second term. Note that n_0 should satisfy

$$d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^6 n \leq d_{\mathcal{X}}^{-\frac{2s}{D}}. \tag{85}$$

Since

$$d_{\mathcal{Y}}^{\frac{4+d_{\mathcal{X}}}{2+d_{\mathcal{X}}}} n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^6 n \leq d_{\mathcal{Y}}^2 n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^6 n \leq d_{\mathcal{X}}^{-\frac{2s}{D}},$$

in the following, we consider solving

$$d_{\mathcal{Y}}^2 n^{-\frac{2}{2+d_{\mathcal{X}}}} \log^6 n \leq d_{\mathcal{X}}^{-\frac{2s}{D}}.$$

Substituting the expression of $d_{\mathcal{X}}$ and $d_{\mathcal{Y}}$, we deduce

$$n^{-\frac{2}{2+\log^{1/2} n}} \log^7 n \leq \log^{-\frac{s}{D}} n \Rightarrow -\frac{2}{2+\log^{\frac{1}{7}2} n} \log n + 7 \log\log n \leq -\frac{s}{D} \log\log n.$$

Denote $a = \log n$. We have

$$\frac{2}{2+a^{1/2}} a \geq \left(7 + \frac{s}{D}\right) \log a. \tag{86}$$

A sufficient condition of (86) is

$$\frac{2}{a^{1/2}} a \geq \left(7 + \frac{s}{D}\right) \log a \Rightarrow a \geq \frac{1}{4} \left(7 + \frac{s}{D}\right)^2 \log^2 a. \tag{87}$$

Note that $\log a < a^{1/3}$ for $a > 100$. Therefore, it is sufficient to solve

$$a \geq \frac{1}{4} \left(7 + \frac{s}{D}\right)^2 a^{\frac{2}{3}} \Rightarrow a \geq \left(\frac{7}{2} + \frac{s}{2D}\right)^6.$$

Substituting a by $\log n$, one has

$$n \geq \exp\left(\max\left\{100, \left(\frac{7}{2} + \frac{s}{2D}\right)^6\right\}\right).$$

Appendix B. Proof of Lemma 6

Proof of Lemma 6. We first prove (18):

$$\begin{aligned}\|E_{\mathcal{H},d}(u) - E_{\mathcal{H},d}(\widetilde{u})\|_2^2 &= \left\|[\langle u-\widetilde{u},\phi_1\rangle_{\mathcal{H}},...,\langle u-\widetilde{u},\phi_d\rangle_{\mathcal{H}}]^\top\right\|_2^2 \\ &= \sum_{k=1}^{d} |\langle u-\widetilde{u},\phi_k\rangle_{\mathcal{H}}|^2 \\ &\leq \sum_{k=1}^{\infty} |\langle u-\widetilde{u},\phi_k\rangle_{\mathcal{H}}|^2 \\ &= \|u-\widetilde{u}\|_{\mathcal{H}}^2.\end{aligned}$$

For (19), we have

$$\|D_{\mathcal{H},d}(\mathbf{a}) - D_{\mathcal{H},d}(\widetilde{\mathbf{a}})\|_{\mathcal{H}}^2 = \left\|\sum_{k=1}^{d}(a_k-\widetilde{a}_k)\phi_k\right\|_{\mathcal{H}}^2 = \|\mathbf{a}-\widetilde{\mathbf{a}}\|_2^2,$$

since $\{\phi_k\}_{k=1}^d$ is an orthonormal set. ■

Appendix C. Proof of Lemma 21

Proof of Lemma 21. Let $\mathbf{a},\widetilde{\mathbf{a}} \in \mathbb{R}^{d_{\mathcal{X}}}$. We have

$$\begin{aligned}\|\Gamma_d^n(\mathbf{a}) - \Gamma_d^n(\widetilde{\mathbf{a}})\|_2 =& \left\|E_{\mathcal{Y}}^n \circ \Psi \circ D_{\mathcal{X}}^n(\mathbf{a}) - E_{\mathcal{Y}}^n \circ \Psi \circ D_{\mathcal{X}}^n(\widetilde{\mathbf{a}})\right\|_2 \\ \leq& L_{E_{\mathcal{Y}}^n} \|\Psi \circ D_{\mathcal{X}}^n(\mathbf{a}) - \Psi \circ D_{\mathcal{X}}^n(\widetilde{\mathbf{a}})\|_2 \\ \leq& L_{E_{\mathcal{Y}}^n} L_\Psi \|D_{\mathcal{X}}^n(\mathbf{a}) - D_{\mathcal{X}}^n(\widetilde{\mathbf{a}})\|_{\mathcal{Y}} \\ \leq& L_{E_{\mathcal{Y}}^n} L_{D_{\mathcal{X}}^n} L_\Psi \|\mathbf{a}-\widetilde{\mathbf{a}}\|_2.\end{aligned}$$

■

Appendix D. Proof of Lemma 24

Proof of Lemma 24. We prove Lemma 24 using the covering number of $\mathcal{F}_{\text{NN}}$. Let $\mathcal{F}^* = \left\{\Gamma_j^*\right\}_{j=1}^{\mathcal{N}(\delta,\mathcal{F}_{\text{NN}},\|\cdot\|_\infty)}$ be a δ-cover of $\mathcal{F}_{\text{NN}}$, where $\mathcal{N}(\delta,\mathcal{F}_{\text{NN}},\|\cdot\|_\infty)$ is the covering number. Then there exists $\Gamma^* \in \mathcal{F}^*$ satisfying $\|\Gamma^* - \Gamma_{\text{NN}}\|_\infty \leq \delta$, where Γ_{NN} is our estimator in (6). Denote $\|\Gamma \circ E_{\mathcal{X}}^n\|_n^2 = \frac{1}{n}\sum_{i=n+1}^{2n} \|\Gamma \circ E_{\mathcal{X}}^n(u_i)\|_2^2$. We have

$$\begin{aligned}&\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\langle \Gamma_{\text{NN}} \circ E_{\mathcal{X}}^n(u_i), \boldsymbol{\epsilon}_i\rangle\right] \\ =&\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\langle \Gamma_{\text{NN}} \circ E_{\mathcal{X}}^n(u_i) - \Gamma^* \circ E_{\mathcal{X}}^n(u_i) + \Gamma^* \circ E_{\mathcal{X}}^n(u_i) - \Gamma_d^n \circ E_{\mathcal{X}}^n(u_i), \boldsymbol{\epsilon}_i\rangle\right]\end{aligned}$$

$$
\begin{aligned}
\leq & \mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\langle\Gamma^*\circ E_{\mathcal{X}}^n(u_i)-\Gamma_d^n\circ E_{\mathcal{X}}^n(u_i),\boldsymbol{\epsilon}_i\rangle\right]+\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\sum_{i=n+1}^{2n}\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i)-\Gamma^*\circ E_{\mathcal{X}}^n(u_i)\|_2\|\boldsymbol{\epsilon}_i\|_2\right]\\
\leq & \mathbb{E}_{\mathcal{S}_2}\left[\frac{\|\Gamma^*\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\|_n}{\sqrt{n}}\frac{\sum_{i=n+1}^{2n}\langle\Gamma^*\circ E_{\mathcal{X}}^n(u_i)-\Gamma_d^n\circ E_{\mathcal{X}}^n(u_i),\boldsymbol{\epsilon}_i\rangle}{\sqrt{n}\left\|\Gamma^*\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\right\|_n}\right]+d_{\mathcal{Y}}\sigma\delta\\
\leq & \sqrt{2}\mathbb{E}_{\mathcal{S}_2}\left[\frac{\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\|_n+\sqrt{d_{\mathcal{Y}}}\delta}{\sqrt{n}}\left|\frac{\sum_{i=n+1}^{2n}\langle\Gamma^*\circ E_{\mathcal{X}}^n(u_i)-\Gamma_d^n\circ E_{\mathcal{X}}^n(u_i),\boldsymbol{\epsilon}_i\rangle}{\sqrt{n}\left\|\Gamma^*\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\right\|_n}\right|\right]+d_{\mathcal{Y}}\sigma\delta,
\end{aligned}
\tag{88}
$$

where the first inequality follows from Cauchy–Schwarz inequality, the third inequality holds since

$$
\begin{aligned}
&\|\Gamma^*\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\|_n\\
=&\sqrt{\frac{1}{n}\sum_{i=n+1}^{2n}\left\|\Gamma^*\circ E_{\mathcal{X}}^n(u_i)-\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i)+\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i)-\Gamma_d^n\circ E_{\mathcal{X}}^n(u_i)\right\|_2^2}\\
\leq&\sqrt{\frac{2}{n}\sum_{i=n+1}^{2n}\left\|\Gamma^*\circ E_{\mathcal{X}}^n(u_i)-\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i)\right\|_2^2+\left\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i)-\Gamma_d^n\circ E_{\mathcal{X}}^n(u_i)\right\|_2^2}\\
\leq&\sqrt{\frac{2}{n}\sum_{i=n+1}^{2n}d_{\mathcal{Y}}\delta^2+\left\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n(u_i)-\Gamma_d^n\circ E_{\mathcal{X}}^n(u_i)\right\|_2^2}\\
\leq&\sqrt{2}\left\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\right\|_n+\sqrt{2d_{\mathcal{Y}}}\delta.
\end{aligned}
$$

Recall that $\left\{\Gamma_j^*\right\}_{j=1}^{\mathcal{N}(\delta,\mathcal{F}_{\mathrm{NN}},\|\cdot\|_\infty)}$ is a δ-cover of $\mathcal{F}_{\mathrm{NN}}$. Denote $z_j=\frac{\sum_{i=n+1}^{2n}\left\langle\Gamma_j^*\circ E_{\mathcal{X}}^n(u_i)-\Gamma_d^n\circ E_{\mathcal{X}}^n(u_i),\boldsymbol{\epsilon}_i\right\rangle}{\sqrt{n}\left\|\Gamma_j^*\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\right\|_n}$. The expectation term in (88) can be bounded as

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{S}_2}\left[\frac{\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\|_n+\sqrt{d_{\mathcal{Y}}}\delta}{\sqrt{n}}\left|\frac{\sum_{i=n+1}^{2n}\langle\Gamma^*\circ E_{\mathcal{X}}^n(u_i)-\Gamma_d^n\circ E_{\mathcal{X}}^n(u_i),\boldsymbol{\epsilon}_i\rangle}{\sqrt{n}\left\|\Gamma^*\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\right\|_n}\right|\right]\\
\leq&\mathbb{E}_{\mathcal{S}_2}\left[\frac{\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\|_n+\sqrt{d_{\mathcal{Y}}}\delta}{\sqrt{n}}\max_j|z_j|\right]\\
\leq&\sqrt{\mathbb{E}_{\mathcal{S}_2}\left[\left(\left\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\right\|_n+\sqrt{d_{\mathcal{Y}}}\delta\right)^2\right]\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{n}\max_j|z_j|^2\right]}\\
\leq&\left(\sqrt{\mathbb{E}_{\mathcal{S}_2}\left[\left(\left\|\Gamma_{\mathrm{NN}}\circ E_{\mathcal{X}}^n-\Gamma_d^n\circ E_{\mathcal{X}}^n\right\|_n\right)^2\right]}+\sqrt{d_{\mathcal{Y}}}\delta\right)\sqrt{\frac{1}{n}\mathbb{E}_{\mathcal{S}_2}\left[\max_j|z_j|^2\right]},
\end{aligned}
\tag{89}
$$

where the second inequality comes from Cauchy–Schwarz inequality, the third inequality comes from the inequality $\sqrt{a+b^2}\leq\sqrt{a}+b$ for $a,b\geq 0$.

Since $\boldsymbol{\epsilon}_i\in[-\sigma,\sigma]^{d_{\mathcal{Y}}}$, each component of $\boldsymbol{\epsilon}_i$ is a sub-Gaussian variable with parameter σ. Therefore for given $u_{n+1},...,u_{2n}$, each z_j is a sub-gaussian variable with parameter $\sqrt{d_{\mathcal{Y}}}\sigma$.

The last term is the maximum of a collection of squared sub-Gaussian variables and is bounded as

$$\begin{aligned}
\mathbb{E}_{\mathcal{S}_2}\left[\max_j |z_j|^2 | u_{n+1}, ..., u_{2n}\right] =& \frac{1}{t}\log\exp\left(t\mathbb{E}_{\mathcal{S}_2}\left[\max_j |z_j|^2 | u_{n+1}, ..., u_{2n}\right]\right) \\
\leq& \frac{1}{t}\log\mathbb{E}_{\mathcal{S}_2}\left[\exp\left(t\max_j |z_j|^2 | u_{n+1}, ..., u_{2n}\right)\right] \\
\leq& \frac{1}{t}\log\mathbb{E}_{\mathcal{S}_2}\left[\sum_j \exp\left(t|z_j|^2 | u_{n+1}, ..., u_{2n}\right)\right] \\
\leq& \frac{1}{t}\log\mathcal{N}\left(\delta, \mathcal{F}_{\mathrm{NN}}, \|\cdot\|_\infty\right) + \frac{1}{t}\log\mathbb{E}_{\mathcal{S}_2}\left[\exp\left(t|z_1|^2 | u_{n+1}, ..., u_{2n}\right)\right].
\end{aligned}$$

Since z_1 is sub-Gaussian with parameter σ^2, we have

$$\begin{aligned}
\mathbb{E}_{\mathcal{S}_2}\left[\exp\left(t|z_1|^2 | u_{n+1}, ..., u_{2n}\right)\right] =& 1 + \sum_{k=1}^{\infty}\frac{t^k\mathbb{E}_{\mathcal{S}_2}\left[z_1^{2k} | u_{n+1}, ..., u_{2n}\right]}{k!} \\
=& 1 + \sum_{k=1}^{\infty}\frac{t^k}{k!}\int_0^{\infty}\mathbb{P}\left(|z_1| \geq \tau^{\frac{1}{2k}} | u_{n+1}, ..., u_{2n}\right)d\tau \\
\leq& 1 + 2\sum_{k=1}^{\infty}\frac{t^k}{k!}\int_0^{\infty}\exp\left(-\frac{\tau^{1/k}}{2d_{\mathcal{Y}}\sigma^2}\right)d\tau \\
=& 1 + \sum_{k=1}^{\infty}\frac{2k(2td_{\mathcal{Y}}\sigma^2)^k}{k!}\Gamma_{\mathrm{G}}(k) \\
=& 1 + 2\sum_{k=1}^{\infty}(2td_{\mathcal{Y}}\sigma^2)^k,
\end{aligned}$$

where Γ_{G} represents the Gamma function. Setting $t = (4d_{\mathcal{Y}}\sigma^2)^{-1}$ gives rise to

$$\begin{aligned}
\mathbb{E}_{\mathcal{S}_2}\left[\max_j |z_j|^2 | u_{n+1}, ..., u_{2n}\right] &\leq 4d_{\mathcal{Y}}\sigma^2\log\mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, \|\cdot\|_\infty) + 4d_{\mathcal{Y}}\sigma^2\log 3 \\
&\leq 4d_{\mathcal{Y}}\sigma^2\log\mathcal{N}(\delta, \mathcal{F}_{\mathrm{NN}}, \|\cdot\|_\infty) + 6d_{\mathcal{Y}}\sigma^2. \qquad (90)
\end{aligned}$$

Substituting (90), (89) into (88) finishes the proof. ■

Appendix E. Proof of Lemma 25

Proof of Lemma 25. Our proof follows the proof of (Chen et al., 2022, Lemma 4.2). Denote $g(u) = \|\Gamma_{\mathrm{NN}} \circ E^n_{\mathcal{X}}(u) - E^n_{\mathcal{Y}} \circ \Psi(u)\|_2^2$. We have $\|g\|_\infty \leq 4d_{\mathcal{Y}}L^2_{E^n_{\mathcal{Y}}}R^2_{\mathcal{Y}}$. Then

$$\mathrm{T}_2 = \mathbb{E}_{\mathcal{S}_2}\left[\mathbb{E}_{u\sim\gamma}\left[g(u)|\mathcal{S}_1\right] - \frac{2}{n}\sum_{i=n+1}^{2n} g(u_i)\right]$$

$$
\begin{aligned}
=&2\mathbb{E}_{\mathcal{S}_2}\left[\frac{1}{2}\mathbb{E}_{u\sim\gamma}\left[g(u)|\mathcal{S}_1\right]-\frac{1}{n}\sum_{i=n+1}^{2n}g(u_i)\right]\\
=&2\mathbb{E}_{\mathcal{S}_2}\left[\mathbb{E}_{u\sim\gamma}[g(u)|\mathcal{S}_1]-\frac{1}{n}\sum_{i=n+1}^{2n}g(u_i)-\frac{1}{2}\mathbb{E}_{u\sim\gamma}\left[g(u)|\mathcal{S}_1\right]\right].
\end{aligned}
\tag{91}
$$

A lower bound of $\frac{1}{2}\mathbb{E}_{u\sim\gamma}\left[g(u)|\mathcal{S}_1\right]$ can be derived as

$$
\mathbb{E}_{u\sim\gamma}\left[g(u)|\mathcal{S}_1\right]=\mathbb{E}_{u\sim\gamma}\left[\frac{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}g(u)|\mathcal{S}_1\right]\geq\frac{1}{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}\mathbb{E}_{u\sim\gamma}\left[g^2(u)|\mathcal{S}_1\right].
\tag{92}
$$

Substituting (92) into (91) gives

$$
\mathrm{T}_2\leq 2\mathbb{E}_{\mathcal{S}_2}\left[\mathbb{E}_{u\sim\gamma}[g(u)|\mathcal{S}_1]-\frac{1}{n}\sum_{i=n+1}^{2n}g(u_i)-\frac{1}{8d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}\mathbb{E}_{u\sim\gamma}\left[g^2(u)|\mathcal{S}_1\right]\right].
$$

Define the set

$$
\mathcal{R}=\left\{g(u)=\|\Gamma\circ E_{\mathcal{X}}^n(u)-E_{\mathcal{Y}}^n\circ\Psi(u)\|_2^2:\Gamma\in\mathcal{F}_{\mathrm{NN}}\right\}.
$$

Denote $\mathcal{S}_2'=\{u_i'\}_{i=n+1}^{2n}$ as an independent copy of $\mathcal{S}_2$. We rewrite T_2 as

$$
\begin{aligned}
\mathrm{T}_2\leq&2\mathbb{E}_{\mathcal{S}_2}\left[\sup_{g\in\mathcal{R}}\left(\mathbb{E}_{\mathcal{S}_2'}\left[\frac{1}{n}\sum_{i=n+1}^{2n}g(u_i')\right]-\frac{1}{n}\sum_{i=n+1}^{2n}g(u_i)-\frac{1}{8d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}\left(\mathbb{E}_{\mathcal{S}_2'}\left[\frac{1}{n}\sum_{i=n+1}^{2n}g^2(u_i')\right]\right)\right)\right]\\
\leq&2\mathbb{E}_{\mathcal{S}_2}\left[\sup_{g\in\mathcal{R}}\left(\mathbb{E}_{\mathcal{S}_2'}\left[\frac{1}{n}\sum_{i=n+1}^{2n}(g(u_i')-g(u_i))\right]-\frac{1}{16d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[\frac{1}{n}\sum_{i=n+1}^{2n}(g^2(u_i)+g^2(u_i'))\right]\right)\right]\\
\leq&2\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[\sup_{g\in\mathcal{R}}\left(\frac{1}{n}\sum_{i=n+1}^{2n}\left((g(u_i)-g(\bar{u}_i))-\frac{1}{16d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[g^2(u_i)+g^2(u_i')\right]\right)\right)\right].
\end{aligned}
\tag{93}
$$

Let $\mathcal{R}^*=\{g_i^*\}_{i=1}^{\mathcal{N}(\delta,\mathcal{R},\|\cdot\|_\infty)}$ be a δ-cover of $\mathcal{R}$. Then for any $g\in\mathcal{R}$, there exists $g^*\in\mathcal{R}^*$ such that $\|g-g^*\|_\infty\leq\delta$.

We next bound (93) using g^*'s. For the first term in (93), we have

$$
\begin{aligned}
g(u_i)-g(u_i')=&g(u_i)-g^*(u_i)+g^*(u_i)-g^*(u_i')+g^*(u_i')-g(u_i')\\
=&\left(g(u_i)-g^*(u_i)\right)+\left(g^*(u_i)-g^*(u_i')\right)+\left(g^*(u_i')-g(u_i')\right)\\
\leq&\left(g^*(u_i)-g^*(u_i')\right)+2\delta.
\end{aligned}
\tag{94}
$$

We lower bound $g^2(u_i)+g^2(u_i')$ as

$$
\begin{aligned}
g^2(u_i)+g^2(u_i')=&\left(g^2(u_i)-(g^*)^2(u_i)\right)+\left((g^*)^2(u_i)+(g^*)^2(u_i')\right)-\left((g^*)^2(u_i')-g^2(u_i')\right)\\
\geq&(g^*)^2(u_i)+(g^*)^2(u_i')-|g(u_i)-g^*(u_i)|\,|g(u_i)+g^*(u_i)|
\end{aligned}
$$

$$
\begin{aligned}
&- \left|g^*(u_i') - g(u_i')\right| \left|g^*(u_i') + g(u_i')\right| \\
\geq &(g^*)^2(u_i) + (g^*)^2(u_i') - 16 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2 \delta. \qquad (95)
\end{aligned}
$$

Substituting (94) and (95) into (93) gives rise to

$$
\begin{aligned}
&\mathrm{T}_2 \\
\leq &2\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[\sup_{g^*\in\mathcal{R}^*}\left(\frac{1}{n}\sum_{i=n+1}^{2n}\left((g^*(u_i) - g^*(u_i')) - \frac{1}{16 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2}\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[(g^*)^2(u_i) + (g^*)^2(u_i')\right]\right)\right)\right] + 6\delta \\
= &2\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[\max_j\left(\frac{1}{n}\sum_{i=n+1}^{2n}\left((g_j^*(u_i) - g_j^*(u_i')) - \frac{1}{16 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2}\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[(g_j^*)^2(u) + (g_j^*)^2(u_i')\right]\right)\right)\right] + 6\delta.
\end{aligned}
$$

Denote $h_j = (u_i, u_i', \xi_i) = (g_j^*(u_i) - g_j^*(u_i'))$. We have

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}[h_j(u_i, u_i')] = 0, \\
&\mathrm{Var}[h_j(u_i, u_i')] = \mathbb{E}\left[h_j^2(u_i, u_i')\right] \\
&\qquad\qquad = \mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[(g_j^*(u_i) - g_j^*(u_i'))^2\right] \\
&\qquad\qquad \leq 2\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[(g_j^*)^2(u_i) + (g_j^*)^2(u_i')\right].
\end{aligned}
$$

Thus T_2 can be bounded as

$$
\begin{aligned}
&\mathrm{T}_2 \leq \widetilde{\mathrm{T}}_2 + 6\delta \\
&\text{with } \widetilde{\mathrm{T}}_2 = 2\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[\max_j\left(\frac{1}{n}\sum_{i=n+1}^{2n}\left(h_j(u_i, u_i') - \frac{1}{32 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2}\mathrm{Var}[h_j(u_i, u_i')]\right)\right)\right].
\end{aligned}
$$

Note that $\|h_j\|_\infty \leq 4 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2$. We next derive the moment generating function of h_j. For any $0 < t < \frac{3}{4 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2}$, we have

$$
\begin{aligned}
\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[\exp(t h_j(u_i, u_i'))\right] = &\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[1 + t h_j(u_i, u_i') + \sum_{k=2}^{\infty}\frac{t^k h_j^k(u_i, u_i')}{k!}\right] \\
\leq &\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[1 + t h_j(u_i, u_i') + \sum_{k=2}^{\infty}\frac{(4 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2)^{k-2} t^k h_j^2(u_i, u_i')}{2 \times 3^{k-2}}\right] \\
= &\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[1 + t h_j(u_i, u_i') + \frac{t^2 h_j^2(u_i, u_i')}{2}\sum_{k=2}^{\infty}\frac{(4 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2)^{k-2} t^{k-2}}{3^{k-2}}\right] \\
= &\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[1 + t h_j(u_i, u_i') + \frac{t^2 h_j^2(u_i, u_i')}{2}\frac{1}{1 - 4 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2 t/3}\right] \\
= &1 + t^2\,\mathrm{Var}[h_j(u_i, u_i')]\frac{1}{2 - 8 d_{\mathcal{Y}} L_{E_{\mathcal{Y}}^n}^2 R_{\mathcal{Y}}^2 t/3}
\end{aligned}
$$

$$\leq \exp\left(\mathrm{Var}[h_j(u_i,u_i')]\frac{3t^2}{6-8d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2t}\right), \tag{96}$$

where the last inequality comes from $1+x\leq\exp(x)$ for $x\geq 0$.

Then for $0<t/n<\frac{3}{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}$, we have

$$\begin{aligned}
&\exp\left(\frac{t\widetilde{\mathrm{T}}_2}{2}\right)\\
=&\exp\left(t\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[\max_j\left(\frac{1}{n}\sum_{i=n+1}^{2n}h_j(u_i,u_i')-\frac{1}{32d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}\frac{1}{n}\sum_{i=n+1}^{2n}\mathrm{Var}[h_j(u_i,u_i')]\right)\right]\right)\\
\leq&\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[\exp\left(t\max_j\left(\frac{1}{n}\sum_{i=n+1}^{2n}h_j(u_i,u_i')-\frac{1}{32d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}\frac{1}{n}\sum_{i=n+1}^{2n}\mathrm{Var}[h_j(u_i,u_i')]\right)\right)\right]\\
\leq&\mathbb{E}_{\mathcal{S}_2,\mathcal{S}_2'}\left[\sum_j\exp\left(\frac{t}{n}\sum_{i=n+1}^{2n}h_j(u_i,u_i')-\frac{t}{32d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}\frac{1}{n}\sum_{i=n+1}^{2n}\mathrm{Var}[h_j(u_i,u_i')]\right)\right]\\
\leq&\left[\sum_j\exp\left(\sum_{i=n+1}^{2n}\mathrm{Var}[h_j(u_i,u_i')]\frac{3t^2/n^2}{6-8d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2t/n}-\frac{1}{32d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}\frac{t}{n}\mathrm{Var}[h_j(u_i,u_i')]\right)\right]\\
=&\left[\sum_j\exp\left(\sum_{i=n+1}^{2n}\frac{t}{n}\mathrm{Var}[h_j(u_i,u_i')]\left(\frac{3t/n}{6-8d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2t/n}-\frac{1}{32d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}\right)\right)\right],
\end{aligned} \tag{97}$$

where the first inequality follows from Jensen's inequality and the third inequality uses (96). Setting

$$\frac{3t/n}{6-8d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2t/n}-\frac{1}{32d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}=0$$

gives $t=\frac{3n}{52d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}<\frac{3n}{4d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}$. Substituting our choice of t into (97) gives

$$\frac{t\widetilde{\mathrm{T}}_2}{2}\leq\log\sum_j\exp(0).$$

Therefore

$$\widetilde{\mathrm{T}}_2\leq\frac{2}{t}\log\mathcal{N}(\delta,\mathcal{R},\|\cdot\|_\infty)=\frac{104d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}{3n}\log\mathcal{N}(\delta,\mathcal{R},\|\cdot\|_\infty)$$

and

$$\mathrm{T}_2\leq\frac{104d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}{3n}\log\mathcal{N}(\delta,\mathcal{R},\|\cdot\|_\infty)+6\delta\leq\frac{35d_{\mathcal{Y}}L_{E_{\mathcal{Y}}^n}^2R_{\mathcal{Y}}^2}{n}\log\mathcal{N}(\delta,\mathcal{R},\|\cdot\|_\infty)+6\delta.$$

We next derive a relation between the covering number of $\mathcal{F}_{\mathrm{NN}}$ and $\mathcal{R}$. For any $g,\widetilde{g}\in\mathcal{R}$, we have

$$g(u)=\left\|\Gamma\circ E_{\mathcal{X}}^n(u)-E_{\mathcal{Y}}^n\circ\Psi(u)\right\|_2^2,\ \widetilde{g}(u)=\left\|\widetilde{\Gamma}\circ E_{\mathcal{X}}^n(u)-E_{\mathcal{Y}}^n\circ\Psi(u)\right\|_2^2$$

for some $\Gamma, \widetilde{\Gamma} \in \mathcal{F}_{\text{NN}}$. We have

$$\begin{aligned}
\|g - \widetilde{g}\|_\infty =& \sup_u \left| \left\| \Gamma \circ E^n_{\mathcal{X}}(u) - E^n_{\mathcal{Y}} \circ \Psi(u) \right\|_2^2 - \left\| \widetilde{\Gamma} \circ E^n_{\mathcal{X}}(u) - E^n_{\mathcal{Y}} \circ \Psi(u) \right\|_2^2 \right| \\
=& \sup_u \left| \left\langle \Gamma \circ E^n_{\mathcal{X}}(u) - \widetilde{\Gamma} \circ E^n_{\mathcal{X}}(u), \Gamma \circ E^n_{\mathcal{X}}(u) + \widetilde{\Gamma} \circ E^n_{\mathcal{X}}(u) - 2E^n_{\mathcal{Y}} \circ \Psi(u) \right\rangle \right| \\
\leq& \sup_u \left\| \Gamma \circ E^n_{\mathcal{X}}(u) - \widetilde{\Gamma} \circ E^n_{\mathcal{X}}(u) \right\|_2 \left\| \Gamma \circ E^n_{\mathcal{X}}(u) + \widetilde{\Gamma} \circ E^n_{\mathcal{X}}(u) - 2E^n_{\mathcal{Y}} \circ \Psi(u) \right\|_2 \\
\leq& 4 d_{\mathcal{Y}} L_{E^n_{\mathcal{Y}}} R_{\mathcal{Y}} \left\| \Gamma - \widetilde{\Gamma} \right\|_\infty .
\end{aligned}$$

As a result, we have

$$\mathcal{N}(\delta, \mathcal{R}, \|\cdot\|_\infty) \leq \mathcal{N}\left(\frac{\delta}{4 d_{\mathcal{Y}} L_{E^n_{\mathcal{Y}}} R_{\mathcal{Y}}}, \mathcal{F}_{\text{NN}}, \|\cdot\|_\infty \right).$$

and Lemma 25 is proved. ■

Appendix F. Proof of Lemma 29

Lemma 29 can be proved similarly to Lemma 25. Denote $g(u) = \left\| \Gamma_{\text{NN}} \circ E^n_{\mathcal{X}}(u) - E^n_{\mathcal{Y}} \circ \Psi(u) \right\|_2^2$ and let $\mathcal{S}_2' = \{u_i'\}_{i=n+1}^{2n}$ be an independent copy of $\mathcal{S}_2$. Following the proof of Lemma 25 up to (93) and replacing $\mathbb{E}_{u \sim \gamma}[g(u)|\mathcal{S}_1]$ by $\mathbb{E}_{\mathcal{S}_2'}\left[\frac{1}{n} \sum_{i=n+1}^{2n} g(u_i') \right]$, we can derive

$$\mathrm{T}_2 \leq 2\mathbb{E}_{\mathcal{S}_2, \mathcal{S}_2'} \left[\sup_{g \in \mathcal{R}} \left(\frac{1}{n} \sum_{i=n+1}^{2n} (g(u_i) - g(u_i')) - \frac{1}{16 d_{\mathcal{Y}} L^2_{E^n_{\mathcal{Y}}} R^2_{\mathcal{Y}}} \frac{1}{n} \sum_{i=n+1}^{2n} \left(g^2(u_i) + g^2(u_i') \right) \right) \right]. \tag{98}$$

Let $\mathcal{R}^* = \{g_i^*\}_{i=1}^{\mathcal{N}(\delta, \mathcal{R}, 2n)}$ be a δ-cover of $\mathcal{R}$ with respect to the data set $\widetilde{\mathcal{S}} = \{u_i\}_{i=1}^n \cup \{u_i'\}_{i=1}^n$. Then for any $g \in \mathcal{R}$, there exists $g^* \in \mathcal{R}^*$ such that $|g(u) - g^*(u)| \leq \delta, \forall u \in \widetilde{\mathcal{S}}$.

Lemma 29 can be proved by following the rest proof of Lemma 25.

Appendix G. Proof of Lemma 35

The proof of Lemma 35 replies on the perturbation theory of operators on separable Hilbert spaces, which is stated in the following lemma:

Lemma 37 (Proposition 2.1 of Giulini (2017)) *Let $A, \widetilde{A}$ be two compact self-adjoint nonnegative operators on the separable real Hilbert space $\mathcal{H}$. Denote the eigenvalues of A and $\widetilde{A}$ in non-increasing order by $\{\lambda_1, \lambda_2, ...\}$ and $\{\widetilde{\lambda}_1, \widetilde{\lambda}_2, ...\}$, respectively. For some integer $d > 0$, let $\Pi_{\mathcal{H},d}$ and $\widetilde{\Pi}_{\mathcal{H},d}$ be the projectors that project any $u \in \mathcal{H}$ to the space spanned by the eigenfunctions corresponding to the largest d eigenvalues of A and $\widetilde{A}$, respectively. We have*

$$\left\| \Pi_{\mathcal{H},d} - \widetilde{\Pi}_{\mathcal{H},d} \right\|_{\text{HS}} \leq \frac{\sqrt{2} \left\| A - \widetilde{A} \right\|_{\text{HS}}}{\max\left\{ \lambda_d - \lambda_{d+1}, \widetilde{\lambda}_d - \widetilde{\lambda}_{d+1} \right\}}. \tag{99}$$

Proof of Lemma 35. Denote $w = \Psi(u)$. Recall that ζ is the probability measure of $v = \Psi(u) + \widetilde{\epsilon}$. We have

$$\begin{aligned} G_\zeta = \mathbb{E}_{\{v_i\}_{i=1}^n \sim \zeta}\left[G_\zeta^n\right] =& \mathbb{E}_{v\sim\zeta}[v \otimes v] \\ =& \mathbb{E}_{w\sim\Psi_{\#}\gamma, \widetilde{\epsilon}\sim\mu}\left[(w+\widetilde{\epsilon}) \otimes (w+\widetilde{\epsilon})\right] \\ =& \mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[w \otimes w\right] + \mathbb{E}_{\widetilde{\epsilon}\sim\mu}\left[\widetilde{\epsilon} \otimes \widetilde{\epsilon}\right] \\ =& G_{\Psi_{\#}\gamma} + G_\mu, \end{aligned} \tag{100}$$

where the third equality holds since w and $\widetilde{\epsilon}$ are independent and $\mathbb{E}\widetilde{\epsilon} = 0$. Recall that $\Pi_{\mathcal{Y},d_\mathcal{Y}}$ (resp. $\Pi^n_{\mathcal{Y},d_\mathcal{Y}}$) projects any $w \in \mathcal{Y}$ to the space spanned by the first $d_\mathcal{Y}$ principal eigenfunctions of $G_{\Psi_{\#}\gamma}$ (resp. G_ζ^n). We denote by $\widetilde{\Pi}_{\mathcal{Y},d_\mathcal{Y}}$ as the projection that projects any $w \in \mathcal{Y}$ to the space spanned by the first $d_\mathcal{Y}$ principal eigenfunctions of G_ζ. Relation (100) implies that

$$\mathbb{E}_{\{v_i\}_{i=1}^n \sim \zeta}\left[\Pi^n_{\mathcal{Y},d_\mathcal{Y}}\right] = \widetilde{\Pi}_{\mathcal{Y},d_\mathcal{Y}}.$$

We have

$$\mathbb{E}_{v\sim\zeta}\left[\left\|G_\zeta^n - \mathbb{E}_{\{v_i\}_{i=1}^n \sim \zeta}\left[G_\zeta^n\right]\right\|_{\mathrm{HS}}^2\right] \le 4\mathbb{E}_{v\sim\zeta}\left[\|v\|_\mathcal{Y}^4\right] \le 4(R_\mathcal{Y} + \widetilde{\sigma})^4.$$

We deduce that

$$\begin{aligned} &\mathbb{E}_\mathcal{S}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi^n_{\mathcal{Y},d_\mathcal{Y}}(w) - w\right\|_\mathcal{Y}^2\right] \\ =&\mathbb{E}_\mathcal{S}\mathbb{E}_{\widetilde{\epsilon}\sim\mu}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi^n_{\mathcal{Y},d_\mathcal{Y}}(w+\widetilde{\epsilon}) - (w+\widetilde{\epsilon}) - \left[\Pi^n_{\mathcal{Y},d_\mathcal{Y}}(\widetilde{\epsilon}) - \widetilde{\epsilon}\right]\right\|_\mathcal{Y}^2\right] \\ \le&2\mathbb{E}_\mathcal{S}\mathbb{E}_{\widetilde{\epsilon}\sim\mu}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi^n_{\mathcal{Y},d_\mathcal{Y}}(w+\widetilde{\epsilon}) - (w+\widetilde{\epsilon})\right\|_\mathcal{Y}^2\right] + 2\mathbb{E}_\mathcal{S}\mathbb{E}_{\widetilde{\epsilon}\sim\mu}\left[\left\|\left[\Pi^n_{\mathcal{Y},d_\mathcal{Y}}(\widetilde{\epsilon}) - \widetilde{\epsilon}\right]\right\|_\mathcal{Y}^2\right] \\ \le&2\mathbb{E}_\mathcal{S}\mathbb{E}_{v\sim\zeta}\left[\left\|\Pi^n_{\mathcal{Y},d_\mathcal{Y}}(v) - v\right\|_\mathcal{Y}^2\right] + 2\mathbb{E}_{\widetilde{\epsilon}\sim\mu}\left[\|\widetilde{\epsilon}\|_\mathcal{Y}^2\right] \\ \le&2\sqrt{\frac{4(R_\mathcal{Y}+\widetilde{\sigma})^4 d_\mathcal{Y}}{n}} + 2\mathbb{E}_{v\sim\zeta}\left[\left\|\widetilde{\Pi}_{\mathcal{Y},d_\mathcal{Y}}(v) - v\right\|_\mathcal{Y}^2\right] + 2\widetilde{\sigma}^2, \end{aligned} \tag{101}$$

where the last inequality comes from Lemma 34 and $\widetilde{\Pi}_{\mathcal{Y},d_\mathcal{Y}} = \mathbb{E}_{v\sim\zeta}[v\otimes v] = \mathbb{E}_{\{v_i\}_{i=1}^n \sim \zeta}\left[\Pi^n_{\mathcal{Y},d_\mathcal{Y}}\right]$.

We bound the second term on the right-hand side as

$$\begin{aligned} &\mathbb{E}_{v\sim\zeta}\left[\left\|\widetilde{\Pi}_{\mathcal{Y},d_\mathcal{Y}}(v) - v\right\|_\mathcal{Y}^2\right] \\ \le&2\mathbb{E}_{v\sim\zeta}\left[\left\|\widetilde{\Pi}_{\mathcal{Y},d_\mathcal{Y}}(v) - \Pi_{\mathcal{Y},d_\mathcal{Y}}(v)\right\|_\mathcal{Y}^2\right] + 2\mathbb{E}_{v\sim\zeta}\left[\left\|\Pi_{\mathcal{Y},d_\mathcal{Y}}(v) - v\right\|_\mathcal{Y}^2\right] \\ \le&2\mathbb{E}_{v\sim\zeta}\left[\left\|\left(\widetilde{\Pi}_{\mathcal{Y},d_\mathcal{Y}} - \Pi_{\mathcal{Y},d_\mathcal{Y}}\right)(v)\right\|_\mathcal{Y}^2\right] + 2\mathbb{E}_{\widetilde{\epsilon}\sim\mu}\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi_{\mathcal{Y},d_\mathcal{Y}}(w+\widetilde{\epsilon}) - (w+\widetilde{\epsilon})\right\|_\mathcal{Y}^2\right] \\ \le&2\mathbb{E}_{v\sim\zeta}\left[\left\|\widetilde{\Pi}_{\mathcal{Y},d_\mathcal{Y}} - \Pi_{\mathcal{Y},d_\mathcal{Y}}\right\|_{\mathrm{op}}^2 \|v\|_\mathcal{Y}^2\right] + 4\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi_{\mathcal{Y},d_\mathcal{Y}}(w) - w\right\|_\mathcal{Y}^2\right] + 4\mathbb{E}_{\widetilde{\epsilon}\sim\mu}\left[\left\|\Pi_{\mathcal{Y},d_\mathcal{Y}}(\widetilde{\epsilon}) - \widetilde{\epsilon}\right\|_\mathcal{Y}^2\right] \end{aligned}$$

$$\leq 2\mathbb{E}_{v\sim\zeta}\left[\left\|\widetilde{\Pi}_{\mathcal{Y},d_{\mathcal{Y}}}-\Pi_{\mathcal{Y},d_{\mathcal{Y}}}\right\|_{\mathrm{HS}}^2\|v\|_{\mathcal{Y}}^2\right]+4\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}(w)-w\right\|_{\mathcal{Y}}^2\right]+4\mathbb{E}_{\widetilde{\epsilon}\sim\mu}\left[\left\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}(\widetilde{\epsilon})-\widetilde{\epsilon}\right\|_{\mathcal{Y}}^2\right]$$

$$\leq 2\left(\frac{\sqrt{2}\|G_\mu\|_{\mathrm{HS}}}{\lambda_{d_{\mathcal{Y}}}-\lambda_{d_{\mathcal{Y}+1}}}\right)^2(R_{\mathcal{Y}}+\widetilde{\sigma})^2+4\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}(w)-w\right\|_{\mathcal{Y}}^2\right]+4\widetilde{\sigma}^2$$

$$\leq 4\left(\frac{\widetilde{\sigma}}{\lambda_{d_{\mathcal{Y}}}-\lambda_{d_{\mathcal{Y}+1}}}\right)^2\widetilde{\sigma}^2(R_{\mathcal{Y}}+\widetilde{\sigma})^2+4\mathbb{E}_{w\sim\Psi_{\#}\gamma}\left[\left\|\Pi_{\mathcal{Y},d_{\mathcal{Y}}}(w)-w\right\|_{\mathcal{Y}}^2\right]+4\widetilde{\sigma}^2, \tag{102}$$

where the fourth inequality follows from Lemma 37.

Substituting (102) into (101) gives rise to (81).

■

Appendix H. Proof of Lemma 36

Proof of Lemma 36. Our proof relies on concepts related to functions on manifolds, such as charts, atlas, the partition of unity, and functions on manifolds. We refer the readers to (Tu, 2011; Lee, 2006; Chen et al., 2022; Liu et al., 2021) for details. Following (Chen and Chen, 1995, Proof of Theorem 1), we first construct an atlas of $\mathcal{M}$ in which all projections projects any point on $\mathcal{M}$ to a tangent space of $\mathcal{M}$. These projections are linear functions that can be realized by a subnetwork. Then the function f is decomposed using a partition of unity subordinates to the atlas we constructed. For each chart (U,ϕ), we use a subnetwork to approximate an indicator function that determines whether the input $\mathbf{x}\in\mathcal{M}$ belongs to U. Another subnetwork is used to approximate $f(\mathbf{x})\circ\phi^{-1}$ on its tangent space. Finally, we multiply both subnetworks together and sum over all chats. The multiplication is approximated by another subnetwork. We prove Lemma 36 in four steps.

Step 1. In the first step, we show that there exists an atlas of $\mathcal{M}$, denoted by $\{U_k,\phi_k\}_{k=1}^{C_{\mathcal{M}}}$, such that ϕ_k's are linear projections. Denote $B_r(\mathbf{c})$ as the Euclidean ball in $\mathbb{R}^{d_{\mathcal{X}}}$ centered at $\mathbf{c}$ with radius r. For any given $r>0$, since $\mathcal{M}$ is compact, there exists a set of points $\{\mathbf{c}_k\}_{k=1}^{C_{\mathcal{M}}}$ such that $\mathcal{M}\in\cup_k B_r(\mathbf{c}_k)$. For each $B_r(\mathbf{c}_k)$, denote $U_i=\mathcal{M}\cap B_r(\mathbf{c}_k)$. By setting $r<\tau/2$, we have that U_i is diffeomorphic to a ball in $\mathbb{R}^{d_0}$ (Niyogi et al., 2008). The minimal number of balls is upper bounded by

$$C_{\mathcal{M}}\leq\left\lceil \mathrm{Area}(\mathcal{M})T_d/r^d\right\rceil,$$

where $\mathrm{Area}(\mathcal{M})$ is the area of $\mathcal{M}$ and T_d is the thickness of U_k's (see Chapter 2 of (Conway and Sloane, 2013)).

We next define ϕ_k's. For each $\mathbf{c}_k$, let $\{\mathbf{v}_j^k\}_{j=1}^{d_0}$ be an orthonormal basis of the tangent space of $\mathcal{M}$ at $\mathbf{c}_k$. Define the matrix $V_k=[\mathbf{v}_1^k,...,\mathbf{v}_d^k]$. We set

$$\phi_k(\mathbf{x})=V_k^\top(\mathbf{x}-\mathbf{c}_k).$$

Note that ϕ_k is a linear function which can be realized by a single layer. Then $\{(U_k,\phi_k)\}_{k=1}^{C_{\mathcal{M}}}$ form an atlas of $\mathcal{M}$.

Step 2. In the second step, we design a subnetwork that determines the chart that the input $\mathbf{x}$ belongs to. To determine whether $\mathbf{x} \in U_k$, it is equivalent to check whether the squared distance between $\mathbf{x}$ and $\mathbf{c}_k$ is less than r^2. It can be done by $\mathbb{1}_{[0,r^2]} \circ d_k^2(\mathbf{x})$ where $\mathbb{1}_{[0,r^2]}(a)$ is an indicator function that outputs 1 if $a \in [0, r^2]$, and outputs 0 otherwise. Here $d_k^2(\mathbf{x})$ is the squared distance function defined as

$$d_k^2(\mathbf{x}) = \|\mathbf{x} - \mathbf{c}_k\|_2^2 = \sum_{j=1}^{d_{\mathcal{X}}} (x_j - c_{k,j})^2,$$

where the notations $\mathbf{x} = [x_1, ..., x_{d_{\mathcal{X}}}]^\top$ and $\mathbf{c}_k = [c_{k,1}, ..., c_{k,d_{\mathcal{X}}}]$ are used.

We next approximate both functions by neural networks. To approximate d_k^2, the key issue is to approximate the square function by neural networks, for which we use the following lemma:

Lemma 38 (Lemma 4.2 of Lu et al. (2021)) *For any $B > 0$ and integers $L, p > 0$, there exists a network $\widetilde{\times}$ in $\mathcal{F}_{\mathrm{NN}}(1, L, 9p + 1, B^2)$ with $d_{\mathcal{Y}} = 1$ such that for any $x, y \in [-B, B]$, we have*

$$|\widetilde{\times}(x, y) - xy| \leq 24B^2 p^{-L}.$$

According to Lemma 38, we approximate $d_k^2(\mathbf{x})$ by

$$\widetilde{d}_k^2(\mathbf{x}) = \sum_{j=1}^{d_{\mathcal{X}}} \widetilde{\times}(x_j - c_{k,j}, x_j - c_{k,j}),$$

where $\widetilde{\times} \in \mathcal{F}_{\mathrm{NN}}(1, 4sL_1, 9p_1+1, B^2)$. The approximation error is $\|\widetilde{d}_k - d_k\|_\infty \leq 24 d_{\mathcal{X}} B^2 p_1^{-4sL_1}$.

For $\mathbb{1}_{[0,r]^2}$, we use the following function to approximate it

$$\widetilde{\mathbb{1}}_\Delta(a) = \begin{cases} 1 & a \leq r^2 - \Delta + 24 d_{\mathcal{X}} B^2 p_1^{-4sL_1}, \\ -\frac{1}{\Delta - 48 d_{\mathcal{X}} B^2 p_1^{-4sL_1}} a + \frac{r^2 - 24 d_{\mathcal{X}} B^2 p_1^{-4sL_1}}{\Delta - 48 d_{\mathcal{X}} B^2 p_1^{-4sL_1}} & a \in \left[r^2 - \Delta + 24 d_{\mathcal{X}} B^2 p_1^{-4sL_1}, r^2 - 24 d_{\mathcal{X}} B^2 p_1^{-4sL_1} \right], \\ 0 & a \geq r^2 - 24 d_{\mathcal{X}} B^2 p_1^{-4sL_1}, \end{cases}$$

where $\Delta \geq 24 d_{\mathcal{X}} B^2 p_1^{-4sL_1}$ will be chosen later. We approximate $\widetilde{\mathbb{1}}_\Delta \circ d_k^2(\mathbf{x})$ by $\widetilde{\mathbb{1}}_\Delta \circ \widetilde{d}_k^2(\mathbf{x})$ in which the parameter Δ is the 'width' of the error region: when $\mathbf{x} \notin U_k$, we have $d_k^2(\mathbf{x}) \geq r^2$ and $\widetilde{\mathbb{1}}_\Delta \circ \widetilde{d}_k^2(\mathbf{x}) = 0$; when $\mathbf{x} \in U_k$ and $d_k^2(\mathbf{x}) \leq r^2 - \Delta$, we have $\widetilde{\mathbb{1}}_\Delta \circ \widetilde{d}_k^2(\mathbf{x}) = 1$.

We then realize $\widetilde{\mathbb{1}}_\Delta(a)$ by a subnetwork. Denoting $m_0 = \frac{1}{\Delta - 48 d_{\mathcal{X}} B^2 p_1^{-4sL_1}}$, $m_1 = r^2 - \Delta + 24 d_{\mathcal{X}} B^2 p_1^{-4sL_1}$, $m_2 = r^2 - 24 d_{\mathcal{X}} B^2 p_1^{-4sL_1}$, we rewrite $\widetilde{\mathbb{1}}_\Delta(a)$ as

$$\widetilde{\mathbb{1}}_\Delta(a) = -m_0(\min\{\max\{a, m_1\}, m_2\}) + m_2 m_0.$$

The function above can be realized by a network with one hidden layer:

$$\widetilde{\mathbb{1}}_\Delta(a) = -m_0 \left(m_2 - \mathrm{ReLU}\left[m_2 - (\mathrm{ReLU}(a - m_1) + m_1) \right] \right) + m_2 m_0.$$

Step 3. In this step, we decompose f using a partition of unity of $\mathcal{M}$ and approximate each component by a subnetwork. Let $\{h_k\}_{k=1}^{C_{\mathcal{M}}}$ be a partition of unity of $\mathcal{M}$ such that h_k is supported on U_k. We decompose f as

$$f = \sum_{k=1}^{C_{\mathcal{M}}} h_k f.$$

Note that for each k, $h_k f$ is a function defined on $\mathcal{M}$ supported on U_k, and $(h_k f) \circ \phi_k^{-1}$ is a function defined in $\mathbb{R}^{d_0}$ and supported on $[-2B, 2B]^{d_0}$. The following lemma shows that $h_k f$ is in the same space as f:

Lemma 39 *Suppose Assumption 8 holds. Let $\{U_k, \phi_k\}_{k=1}^{C_{\mathcal{M}}}$ be defined in* **Step 1**. *For each k, we have $h_k f \in \mathcal{C}^{0,1}(\mathcal{M})$ and $\|h_k f\|_{\mathcal{C}^{0,1}(\mathcal{M})}$ is bounded by a constant depending on d_0, h_k, f and ϕ_k.*

Lemma 39 can be proved by following the proof of (Chen et al., 2022, Lemma 2). The proof is omitted here. According to Lemma 39 and since ϕ_k is a linear projection, we have $(h_k f) \circ \phi_k^{-1} \in \mathcal{C}^{0,1}([-2B, 2B]^{d_0})$. Lemma 27 implies that there exists a neural network $\widetilde{f}_k \in \mathcal{F}_{\text{NN}}(1, L_2, p_2, M)$ with

$$L_2 = O(\widetilde{L}_2),\ p_2 = O\left(\widetilde{p}_2\right),\ M = R$$

for any $\widetilde{L}_2, \widetilde{p}_2 > 0$ such that

$$\|\widetilde{f}_k - (h_k f) \circ \phi_k^{-1}\|_\infty \leq C_1 \widetilde{L}_2^{-\frac{2}{d_0}} \widetilde{p}_2^{-\frac{2}{d_0}}$$

for some constant C_1 depending on d_0, B, R.

Step 4. We then assemble all subnetworks constructed in the previous steps and approximate f by

$$\widetilde{f} = \sum_{k=1}^{C_{\mathcal{M}}} \widetilde{\times}\left(\widetilde{f}_k \circ \phi_k, \widetilde{\mathbb{1}}_k \circ \widetilde{d}_k^2\right). \tag{103}$$

In (103), according to Lemma 38, we set $\widetilde{\times} \in \mathcal{F}_{\text{NN}}(1, 4L_3, 9p_3 + 1, M)$ as an approximation of $\times$ with $M = R$ and error $24R^2 p_3^{-4L_3}$. The following lemma gives an upper bound of the approximation error of $\widetilde{f}$ (see a proof in Appendix I):

Lemma 40 *The error of $\widetilde{f}$ can be decomposed as*

$$\|\widetilde{f} - f\|_\infty \leq \sum_{k=1}^{C_{\mathcal{M}}} A_{k,1} + A_{k,2} + A_{k,3}$$

with

$$A_{k,1} = \left\|\widetilde{\times}(\widetilde{f} \circ \phi_k^{-1}, \widetilde{\mathbb{1}}_\Delta \circ \widetilde{d}_k^2) - (\widetilde{f} \circ \phi_k^{-1}) \times (\widetilde{\mathbb{1}}_\Delta \circ \widetilde{d}_k^2)\right\|_\infty \leq 24R^2 p_3^{-2} L_3^{-2},$$

$$A_{k,2} = \left\|(\widetilde{f}\circ\phi_k^{-1})\times(\widetilde{\mathbb{1}}_\Delta\circ\widetilde{d}_k^2) - [(h_kf)\circ\phi_k^{-1}]\times(\widetilde{\mathbb{1}}_\Delta\circ\widetilde{d}_k^2)\right\|_\infty \le C_{12}\widetilde{L}_2^{-\frac{2}{d_0}}\widetilde{p}_2^{-\frac{2}{d_0}},$$
$$A_{k,3} = \left\|[(h_kf)\circ\phi_k^{-1}]\times(\widetilde{\mathbb{1}}_\Delta\circ\widetilde{d}_k^2) - [(h_kf)\circ\phi_k^{-1}]\times\mathbb{1}_{[0,r^2]}\right\|_\infty \le \frac{C_{13}(\pi+1)}{r(1-r/\tau)}\Delta,$$

for some constant C_{12} depending on d_0, τ, B, R, and C_{13} depending on R.

According to Lemma 40, for any $\widetilde{L}, \widetilde{p} > 0$, we set

- $\widetilde{f}_k \in \mathcal{F}_{\rm NN}(1, L_2, p_2, M)$ with $L_2 = O\left(\widetilde{L}\right), p_2 = O\left(\widetilde{p}\right)$,
- $\widetilde{\times} \in \mathcal{F}_{\rm NN}(1, 4L_3, 9p_3+1, M)$ with $L_3 = O\left(\widetilde{L}\right), p_3 = O\left(\widetilde{p}\right)$,
- $\widetilde{d}_k^2 \in \mathcal{F}_{\rm NN}(1, 4L_1, d_{\mathcal{X}}(9p_1+1), M)$ with $\Delta = \widetilde{L}^{-\frac{2}{d_0}}\widetilde{p}^{-\frac{2}{d_0}}$, $L_1 = \widetilde{L} + \log(12d_{\mathcal{X}}B^2), p_1 = \widetilde{p}$ such that

$$\begin{aligned}
&24d_{\mathcal{X}}B^2p_1^{-4L_1} = 24d_{\mathcal{X}}B^2\widetilde{p}^{-4\widetilde{L}-\log(48d_{\mathcal{X}}B^2)} = 24d_{\mathcal{X}}B^2\widetilde{p}^{-\log(48d_{\mathcal{X}}B^2)}\widetilde{p}^{-4\widetilde{L}}\\
&= 24d_{\mathcal{X}}B^2(48d_{\mathcal{X}}B^2)^{-\log\widetilde{p}}\widetilde{p}^{-4\widetilde{L}} \le 24d_{\mathcal{X}}B^2(24d_{\mathcal{X}}B^2)^{-1}\widetilde{p}^{-4\widetilde{L}}\\
&\le \widetilde{p}^{-4\widetilde{L}} \le \widetilde{p}^{-2(\widetilde{L}+1)} \le \widetilde{p}^{-2}2^{-2\widetilde{L}} \le \widetilde{p}^{-2}\widetilde{L}^{-2} < \Delta,
\end{aligned}$$

  where in the third equality, we used $a^{\log b} = b^{\log a}$ for $a, b > 0$,

- $\widetilde{\mathbb{1}}_\Delta \in \mathcal{F}_{\rm NN}(1, 2, 1, 1)$.

The total approximation error is bounded by $C_3\widetilde{L}^{-\frac{2}{d_0}}\widetilde{p}^{-\frac{2}{d_0}}$ for some C_3 depending on d_0, R, B, τ and the surface area of $\mathcal{M}$. The constant hidden in $O(\cdot)$ depends on d_0, R, B, τ and the surface area of $\mathcal{M}$. The resulting network is in $\mathcal{F}_{\rm NN}(1, L, p, M)$ with L, p, M defined in (82).

■

Appendix I. Proof of Lemma 40

Proof of Lemma 40. For $A_{k,1}$, since $\widetilde{\times} \in \mathcal{F}_{\rm NN}(1, 4L_3, p_3, R)$, by Lemma 38, we have

$$A_{k,1} \le 24R^2p_3^{-4L_3} \le 24R^2p_3^{-2(L_3+1)} \le 24R^2p_3^{-2}2^{-2L_3} \le 24R^2p_3^{-2}L_3^{-2}.$$

For $A_{k,2}$, since $\widetilde{\mathbb{1}}_k \circ \widetilde{d}_k^2 \in [0,1]$, we have

$$A_{k,2} \le \left\|\widetilde{f}\circ\phi_k^{-1} - (h_kf)\circ\phi_k^{-1}\right\|_\infty \le \left\|\widetilde{f} - (h_kf)\right\|_\infty \le C_{12}\widetilde{L}_2^{-\frac{2}{d_0}}\widetilde{p}_2^{-\frac{2}{d_0}}.$$

The upper bound of $A_{k,3}$ is proved in (Chen et al., 2022, Proof of Lemma 3). ■

References

Anima Anandkumar, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Nikola Kovachki, Zongyi Li, Burigede Liu, and Andrew Stuart. Neural operator: Graph kernel network for partial differential equations. In *ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations*, 2020. URL https://openreview.net/forum?id=fg2ZFmXFO3.

Martin Anthony and P Bartlett. Neural network learning: theoretical foundations, 1999.

A. R. Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory*, 39(3):930–945, May 1993. ISSN 0018-9448. doi: 10.1109/18.256500.

Peter L Bartlett, Olivier Bousquet, and Shahar Mendelson. Local Rademacher complexities. *Annals of Statistics*, 2005.

Peter L Bartlett, Nick Harvey, Christopher Liaw, and Abbas Mehrabian. Nearly-tight VC-dimension and pseudodimension bounds for piecewise linear neural networks. *The Journal of Machine Learning Research*, 20(1):2285–2301, 2019.

Benedikt Bauer and Michael Kohler. On deep learning as a remedy for the curse of dimensionality in nonparametric regression. *The Annals of Statistics*, 47(4):2261 – 2285, 2019. doi: 10.1214/18-AOS1747. URL https://doi.org/10.1214/18-AOS1747.

Julius Berner, Philipp Grohs, and Arnulf Jentzen. Analysis of the generalization error: Empirical risk minimization over deep artificial neural networks overcomes the curse of dimensionality in the numerical approximation of black-scholes partial differential equations. *CoRR*, abs/1809.03062, 2018. URL http://arxiv.org/abs/1809.03062.

Kaushik Bhattacharya, Bamdad Hosseini, Nikola B Kovachki, and Andrew M Stuart. Model reduction and neural networks for parametric PDEs. *The SMAI Journal of Computational Mathematics*, 7:121–157, 2021.

Hans-Joachim Bungartz and Michael Griebel. Sparse grids. *Acta numerica*, 13:147–269, 2004.

Shengze Cai, Zhicheng Wang, Lu Lu, Tamer A. Zaki, and George Em Karniadakis. DeepM&Mnet: Inferring the electroconvection multiphysics fields based on operator approximation by neural networks. *Journal of Computational Physics*, 436:110296, 2021. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2021.110296. URL https://www.sciencedirect.com/science/article/pii/S0021999121001911.

Yuan Cao and Quanquan Gu. Generalization bounds of stochastic gradient descent for wide and deep neural networks. *CoRR*, abs/1905.13210, 2019. URL `http://arxiv.org/abs/1905.13210`.

Long Qing Chen and Jie Shen. Applications of semi-implicit fourier-spectral method to phase field equations. *Computer Physics Communications*, 108(2-3):147–158, 1998.

Minshuo Chen, Haoming Jiang, Wenjing Liao, and Tuo Zhao. Efficient approximation of deep ReLU networks for functions on low dimensional manifolds. *Advances in Neural Information Processing Systems*, 32:8174–8184, 2019.

Minshuo Chen, Hao Liu, Wenjing Liao, and Tuo Zhao. Doubly robust off-policy learning on low-dimensional manifolds by deep neural networks. *arXiv preprint arXiv:2011.01797*, 2020.

Minshuo Chen, Haoming Jiang, Wenjing Liao, and Tuo Zhao. Nonparametric regression on low-dimensional manifolds using deep ReLU networks: Function approximation and statistical recovery. *Information and Inference: A Journal of the IMA*, 11(4):1203–1253, 2022.

Tianping Chen 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(4):911–917, 1995.

Abdellah Chkifa, Albert Cohen, and Christoph Schwab. Breaking the curse of dimensionality in sparse polynomial approximation of parametric PDEs. *Journal de Mathématiques Pures et Appliquées*, 103(2):400–428, 2015.

Alexander Cloninger and Timo Klock. ReLU nets adapt to intrinsic dimensionality beyond the target domain. *arXiv e-prints*, pages arXiv–2008, 2020.

Albert Cohen and Ronald DeVore. Approximation of high-dimensional parametric PDEs. *Acta Numerica*, 24:1–159, 2015.

John Horton Conway and Neil James Alexander Sloane. *Sphere Packings, Lattices and Groups*, volume 290. Springer Science & Business Media, 2013.

George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, 1989.

Maarten V de Hoop, Nikola B Kovachki, Nicholas H Nelsen, and Andrew M Stuart. Convergence rates for learning linear operators from noisy data. *arXiv preprint arXiv:2108.12515*, 2021.

Mo Deng, Shuai Li, Alexandre Goy, Iksung Kang, and George Barbastathis. Learning to synthesize: robust phase retrieval at low photon counts. *Light: Science & Applications*, 9(1):36, 2020. doi: 10.1038/s41377-020-0267-2. URL https://doi.org/10.1038/s41377-020-0267-2.

Qiang Du, Yiqi Gu, Haizhao Yang, and Chao Zhou. The discovery of dynamics via linear multistep methods and deep learning: Error estimation. *arXiv preprint arXiv:2103.11488*, 2021.

Chenguang Duan, Yuling Jiao, Yanming Lai, Xiliang Lu, and Zhijian Yang. Convergence rate analysis for deep Ritz method. *arxiv:2103.13330*, 2021.

Weinan E, Chao Ma, and Lei Wu. A priori estimates of the population risk for two-layer neural networks. *Communications in Mathematical Sciences*, 17(5):1407–1425, 2019.

Weinan E, Chao Ma, and Lei Wu. The barron space and the flow-induced function spaces for neural network models. *Constructive Approximation*, 2021. doi: 10.1007/s00365-021-09549-y. URL https://doi.org/10.1007/s00365-021-09549-y.

Alexandre Ern and Jean-Luc Guermond. *Theory and practice of finite elements*, volume 159. Springer, 2004.

Yuwei Fan, Jordi Feliu-Fabà, Lin Lin, Lexing Ying, and Leonardo Zepeda-Núñez. A multiscale neural network based on hierarchical nested bases. *Research in the Mathematical Sciences*, 6(2):21, 2019a. doi: 10.1007/s40687-019-0183-3. URL https://doi.org/10.1007/s40687-019-0183-3.

Yuwei Fan, Cindy Orozco Bohorquez, and Lexing Ying. BCR-Net: A neural network based on the nonstandard wavelet form. *Journal of Computational Physics*, 384:1–15, 2019b. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2019.02.002. URL https://www.sciencedirect.com/science/article/pii/S0021999119300762.

Max H. Farrell, Tengyuan Liang, and Sanjog Misra. Deep neural networks for estimation and inference. *Econometrica*, 89(1):181–213, 2021. ISSN 0012-9682. doi: 10.3982/ecta16901. URL http://dx.doi.org/10.3982/ECTA16901.

Herbert Federer. Curvature measures. *Transactions of the American Mathematical Society*, 93(3):418–491, 1959.

Ilaria Giulini. Robust PCA and pairs of projections in a Hilbert space. *Electronic Journal of Statistics*, 11(2):3903–3926, 2017.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT press, 2016.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE, 2013.

Yiqi Gu, John Harlim, Senwei Liang, and Haizhao Yang. Stationary density estimation of Itô diffusions using deep learning. *arxiv:2109.03992*, 2021.

László Györfi, Michael Kohler, Adam Krzyżak, and Harro Walk. *A Distribution-free Theory of Nonparametric Regression*, volume 1. Springer, 2002.

Bernard Haasdonk. Reduced basis methods for parametrized PDEs–a tutorial introduction for stationary and instationary problems. *Model Reduction and Approximation: Theory and Algorithms*, 15:65, 2017.

Michael Hamers and Michael Kohler. Nonasymptotic bounds on the L2 error of neural network regression estimates. *Annals of the Institute of Statistical Mathematics*, 58(1): 131–151, 2006.

Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, and Brian Kingsbury. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, 29, 2012.

Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991.

Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417, 1933.

Harold Hotelling. Relations between two sets of variates. In *Breakthroughs in Statistics*, pages 162–190. Springer, 1992.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *CoRR*, abs/1806.07572, 2018. URL `http://arxiv.org/abs/1806.07572`.

Yuling Jiao, Guohao Shen, Yuanyuan Lin, and Jian Huang. Deep nonparametric regression on approximately low-dimensional manifolds. *arXiv: Statistics Theory*, 2021.

Yuehaw Khoo and Lexing Ying. SwitchNet: A neural network model for forward and inverse scattering problems. *SIAM Journal on Scientific Computing*, 41(5):A3182–A3201, 2019. doi: 10.1137/18M1222399.

Yuehaw Khoo, Jianfeng Lu, and Lexing Ying. Solving parametric PDE problems with artificial neural networks. *European Journal of Applied Mathematics*, 32(3):421–435, 2021.

Michael Kohler and Adam Krzyżak. Adaptive regression estimation with multilayer feedforward neural networks. *Nonparametric Statistics*, 17(8):891–913, 2005.

Michael Kohler, Adam Krzyzak, and Sophie Langer. Estimation of a function of low local dimensionality by deep neural networks. *arxiv:1908.11140*, 2020.

Vladimir Koltchinskii. Local Rademacher complexities and oracle inequalities in risk minimization. *Annals of Statistics*, 2006.

Nikola Kovachki, Samuel Lanthaler, and Siddhartha Mishra. On universal approximation and error bounds for Fourier neural operators. *Journal of Machine Learning Research*, 22(290):1–76, 2021.

Nikola B Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew M Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to PDEs. *J. Mach. Learn. Res.*, 24(89):1–97, 2023.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural Information Processing Systems*, pages 1097–1105, 2012.

Samuel Lanthaler and Andrew M Stuart. The curse of dimensionality in operator learning. *arXiv preprint arXiv:2306.15924*, 2023.

Samuel Lanthaler, Siddhartha Mishra, and George E Karniadakis. Error estimates for deeponets: A deep learning framework in infinite dimensions. *Transactions of Mathematics and Its Applications*, 6(1):tnac001, 2022.

John M Lee. *Riemannian Manifolds: An Introduction to Curvature*, volume 176. Springer Science & Business Media, 2006.

Dong Li, Zhonghua Qiao, and Tao Tang. Characterizing the stabilization size for semi-implicit fourier-spectral method to phase field equations. *SIAM Journal on Numerical Analysis*, 54(3):1653–1681, 2016.

Zongyi Li, Nikola Borislavov Kovachki, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, Anima Anandkumar, et al. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2020.

Chensen Lin, Zhen Li, Lu Lu, Shengze Cai, Martin Maxey, and George Em Karniadakis. Operator learning for predicting multiscale bubble growth dynamics. *The Journal of Chemical Physics*, 154(10):104118, 2021. doi: 10.1063/5.0041203.

Hao Liu, Minshuo Chen, Tuo Zhao, and Wenjing Liao. Besov function approximation and binary classification on low-dimensional manifolds using convolutional residual networks. In *International Conference on Machine Learning*, 2021.

Hao Liu, Minshuo Chen, Siawpeng Er, Wenjing Liao, Tong Zhang, and Tuo Zhao. Benefits of overparameterized convolutional residual networks: Function approximation under smoothness constraint. *arXiv preprint arXiv:2206.04569*, 2022.

Hao Liu, Alex Havrilla, Rongjie Lai, and Wenjing Liao. Deep nonparametric estimation of intrinsic data structures by chart autoencoders: Generalization error and robustness. *Applied and Computational Harmonic Analysis*, 68:101602, 2024.

Jianfeng Lu and Yulong Lu. A priori generalization error analysis of two-layer neural networks for solving high dimensional Schrödinger eigenvalue problems. *arxiv:2105.01228*, 2021.

Jianfeng Lu, Yulong Lu, and Min Wang. A priori generalization analysis of the deep Ritz method for solving high dimensional elliptic equations. *arxiv:2101.01708*, 2021.

Jianfeng Lu, Zuowei Shen, Haizhao Yang, and Shijun Zhang. Deep network approximation for smooth functions. *SIAM Journal on Mathematical Analysis*, 2021.

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(3):218–229, 2021. doi: 10.1038/s42256-021-00302-5. URL `https://doi.org/10.1038/s42256-021-00302-5`.

Tao Luo and Haizhao Yang. Two-layer neural networks for partial differential equations: Optimization and generalization theory. *ArXiv*, abs/2006.15733, 2020.

Hrushikesh Mhaskar. Local approximation of operators. *arXiv preprint arXiv:2202.06392*, 2022.

Riccardo Miotto, Fei Wang, Shuang Wang, Xiaoqian Jiang, and Joel T Dudley. Deep learning for healthcare: review, opportunities and challenges. *Briefings in Bioinformatics*, 19(6):1236–1246, 2017.

Siddhartha Mishra and Roberto Molinaro. Estimates on the generalization error of physics informed neural networks (PINNs) for approximating PDEs. *arxiv:2006.16144*, 2020.

Ryumei Nakada and Masaaki Imaizumi. Adaptive approximation and generalization of deep neural network with intrinsic dimensionality. *Journal of Machine Learning Research*, 21 (174):1–38, 2020a. URL `http://jmlr.org/papers/v21/20-002.html`.

Ryumei Nakada and Masaaki Imaizumi. Adaptive approximation and generalization of deep neural network with intrinsic dimensionality. *J. Mach. Learn. Res.*, 21:174–1, 2020b.

Nicholas H Nelsen and Andrew M Stuart. The random feature model for input-output maps between banach spaces. *arXiv preprint arXiv:2005.10224*, 2020.

Partha Niyogi, Stephen Smale, and Shmuel Weinberger. Finding the homology of submanifolds with high confidence from random samples. *Discrete & Computational Geometry*, 39(1-3):419–441, 2008.

Steven A Orszag. Accurate solution of the Orr–Sommerfeld stability equation. *Journal of Fluid Mechanics*, 50(4):689–703, 1971.

Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11): 559–572, 1901.

Benjamin Peherstorfer and Karen Willcox. Data-driven operator inference for non-intrusive projection-based model reduction. *Computer Methods in Applied Mechanics and Engineering*, 306:196–215, 2016. ISSN 0045-7825. doi: https://doi.org/10.1016/j.cma.2016.03.025. URL https://www.sciencedirect.com/science/article/pii/S0045782516301104.

Chang Qiao, Di Li, Yuting Guo, Chong Liu, Tao Jiang, Qionghai Dai, and Dong Li. Evaluation and development of deep neural networks for image super-resolution in optical microscopy. *Nature Methods*, 18(2):194–202, 2021. doi: 10.1038/s41592-020-01048-5. URL https://doi.org/10.1038/s41592-020-01048-5.

Zhen Qin, Qingliang Zeng, Yixin Zong, and Fan Xu. Image inpainting based on deep learning: A review. *Displays*, 69:102028, 2021. ISSN 0141-9382. doi: https://doi.org/10.1016/j.displa.2021.102028. URL https://www.sciencedirect.com/science/article/pii/S0141938221000391.

Gianluigi Rozza. Fundamentals of reduced basis method for problems governed by parametrized PDEs and applications. In *Separated Representations and PGD-based Model Reduction*, pages 153–227. Springer, 2014.

Johannes Schmidt-Hieber. Deep relu network approximation of functions on a manifold. *arXiv preprint arXiv:1908.00695*, 2019.

Johannes Schmidt-Hieber. Nonparametric regression using deep neural networks with ReLU activation function. *The Annals of Statistics*, 48(4):1875–1897, 2020.

Martin H Schultz. L^∞-multivariate approximation theory. *SIAM Journal on Numerical Analysis*, 6(2):161–183, 1969.

Uri Shaham, Alexander Cloninger, and Ronald R Coifman. Provable approximation properties for deep neural networks. *Applied and Computational Harmonic Analysis*, 44(3): 537–557, 2018.

Jie Shen, Tao Tang, and Li-Lian Wang. *Spectral Methods: Algorithms, Analysis and Applications*, volume 41. Springer Science & Business Media, 2011.

Zuowei Shen, Haizhao Yang, and Shijun Zhang. Deep network approximation characterized by number of neurons. *Communications in Computational Physics*, 28(5):1768–1811, 2020. ISSN 1991-7120. doi: 10.4208/cicp.OA-2020-0149.

Zuowei Shen, Haizhao Yang, and Shijun Zhang. Deep network with approximation error being reciprocal of width to power of square root of depth. *Neural Computation*, 33 (4):1005–1036, 03 2021a. ISSN 0899-7667. doi: 10.1162/neco_a_01364. URL `https://doi.org/10.1162/neco_a_01364`.

Zuowei Shen, Haizhao Yang, and Shijun Zhang. Neural network approximation: Three hidden layers are enough. *Neural Networks*, 141:160–173, 2021b. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2021.04.011.

Zuowei Shen, Haizhao Yang, and Shijun Zhang. Deep network approximation: Achieving arbitrary accuracy with fixed number of neurons. *arxiv:2107.02397*, 2021c.

Zuowei Shen, Haizhao Yang, and Shijun Zhang. Optimal approximation rate of ReLU networks in terms of width and depth. *Journal de Mathématiques Pures et Appliquées*, to appear.

Yeonjong Shin, Jerome Darbon, and George Em Karniadakis. On the convergence of physics informed neural networks for linear second-order elliptic and parabolic type PDEs. *arxiv:2004.01806*, 2020.

Jonathan W. Siegel and Jinchao Xu. Sharp bounds on the approximation rates, metric entropy, and n-widths of shallow neural networks. *arxiv:2101.12365*, 2021.

Charles J. Stone. Optimal Global Rates of Convergence for Nonparametric Regression. *The Annals of Statistics*, 10(4):1040 – 1053, 1982. doi: 10.1214/aos/1176345969. URL `https://doi.org/10.1214/aos/1176345969`.

Taiji Suzuki. Adaptivity of deep ReLU network for learning in besov and mixed smooth besov spaces: optimal rate and curse of dimensionality. *arXiv preprint arXiv:1810.08033*, 2018.

Gabor Szeg. *Orthogonal Polynomials*, volume 23. American Mathematical Soc., 1939.

Chunwei Tian, Lunke Fei, Wenxian Zheng, Yong Xu, Wangmeng Zuo, and Chia-Wen Lin. Deep learning on image denoising: An overview. *Neural Networks*, 131:251–275, Nov 2020. ISSN 0893-6080. doi: 10.1016/j.neunet.2020.07.025. URL http://dx.doi.org/10.1016/j.neunet.2020.07.025.

Loring W.. Tu. *An Introduction to Manifolds*. Springer., 2011.

Aad W Van Der Vaart, Adrianus Willem van der Vaart, Aad van der Vaart, and Jon Wellner. *Weak convergence and empirical processes: with applications to statistics*. Springer Science & Business Media, 1996.

Zhun Wei and Xudong Chen. Physics-inspired convolutional neural network for solving full-wave inverse scattering problems. *IEEE Transactions on Antennas and Propagation*, 67(9):6138–6148, 2019.

Dmitry Yarotsky. Error bounds for approximations with deep relu networks. *Neural Networks*, 94:103–114, 2017.

Dmitry Yarotsky. Optimal approximation of continuous functions by very deep ReLU networks. In Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet, editors, *Proceedings of the 31st Conference On Learning Theory*, volume 75 of *Proceedings of Machine Learning Research*, pages 639–649. PMLR, 06–09 Jul 2018.

Dmitry Yarotsky. Elementary superexpressive activations. *arXiv e-prints*, 2021.

Dmitry Yarotsky and Anton Zhevnerchuk. The phase diagram of approximation rates for deep neural networks. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 13005–13015. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/979a3f14bae523dc5101c52120c535e9-Paper.pdf.

Zecheng Zhang, Wing Tat Leung, and Hayden Schaeffer. A discretization-invariant extension and analysis of some deep operator networks. *arXiv preprint arXiv:2307.09738*, 2023a.

Zecheng Zhang, Leung Wing Tat, and Hayden Schaeffer. BelNet: Basis enhanced learning, a mesh-free neural operator. *Proceedings of the Royal Society A*, 479(2276):20230043, 2023b.

Yinhao Zhu and Nicholas Zabaras. Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. *Journal of Computational Physics*, 366:415–447, 2018. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2018.04.018. URL https://www.sciencedirect.com/science/article/pii/S0021999118302341.