DEEP NETWORK APPROXIMATION CHARACTERIZED BY NUMBER OF NEURONS*

ZUOWEI SHEN†, HAIZHAO YANG‡, AND SHIJUN ZHANG§

Abstract. This paper quantitatively characterizes the approximation power of deep feed-forward neural networks (FNNs) in terms of the number of neurons, i.e., the product of the network width and depth. It is shown by construction that ReLU FNNs with width $\max\{8d\lfloor N^{1/d}\rfloor + 4d, 12N + 14\}$ and depth $9L + 12$ can approximate an arbitrary Hölder continuous function of order α with a Lipschitz constant ν on $[0,1]^d$ with a tight approximation rate $5(8\sqrt{d})^{\alpha}\nu N^{-2\alpha/d}L^{-2\alpha/d}$ for any given $N, L \in \mathbb{N}^+$. The constructive approximation is a corollary of a more general result for an arbitrary continuous function f in terms of its modulus of continuity $\omega_f(\cdot)$. In particular, the approximation rate of ReLU FNNs with width $\max\{8d\lfloor N^{1/d}\rfloor + 4d, 12N + 14\}$ and depth $9L + 12$ for a general continuous function f is $5\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d})$. We also extend our analysis to the case when the domain of f is irregular or localized in an ε-neighborhood of a $d_{\mathcal{M}}$-dimensional smooth manifold $\mathcal{M} \subseteq [0,1]^d$ with $d_{\mathcal{M}} \ll d$. Especially, in the case of an essentially low-dimensional domain, we show an approximation rate $3\omega_f\big(\frac{4\varepsilon}{1-\delta}\sqrt{\frac{d}{d_\delta}}\big) + 5\omega_f\big(\frac{16d}{(1-\delta)\sqrt{d_\delta}}N^{-2/d_\delta}L^{-2/d_\delta}\big)$ for ReLU FNNs to approximate f in the ε-neighborhood, where $d_\delta = \mathcal{O}\big(d_{\mathcal{M}}\frac{\ln(d/\delta)}{\delta^2}\big)$ for any given $\delta \in (0,1)$. Our analysis provides a general guide for selecting the width and the depth of ReLU FNNs to approximate continuous functions especially in parallel computing.

Key words. Deep ReLU Neural Networks, Hölder Continuity, Modulus of Continuity, Approximation Theory, Low-Dimensional Manifold, Parallel Computing.

1. Introduction.

The approximation theory of neural networks has been an active research topic in the past few decades. Previously, as a special kind of ridge function approximation, shallow neural networks with one hidden layer and various activation functions (e.g., wavelets pursuits [42, 10], adaptive splines [19, 50], radial basis functions [8, 48, 18, 23, 60], sigmoid functions [27, 41, 34, 7, 38, 35, 14, 13, 15]) were widely discussed and admit good approximation properties, e.g., the universal approximation property [16, 28, 27], overcoming the curse of dimensionality [4], and providing attractive approximation rate in nonlinear approximation [19, 42, 10, 50, 18, 23, 60].

The introduction of deep networks with more than one hidden layers has made significant impacts in many fields in computer science and engineering including computer vision [32] and natural language processing [1]. New scientific computing tools based on deep networks have also emerged and facilitated large-scale and high-dimensional problems that were impractical previously [22, 20]. The design of deep ReLU networks is the key of such a revolution. These breakthroughs have stimulated broad research topics from different points of views to study the power of deep ReLU networks, e.g. in terms of combinatorics [46], topology [6], Vapnik-Chervonenkis (VC) dimension [5, 53, 25], fat-shattering dimension [31, 2], information theory [49], classical approximation theory [16, 28, 4, 62, 57, 57], optimization [29, 47, 30] etc.

Particularly in approximation theory, **non-quantitative and asymptotic** approximation rates of ReLU FNNs have been proposed for various types of functions. For example, smooth functions [39, 36, 61, 21], piecewise smooth functions [49], band-limited functions [45], continuous functions [62]. However, to the best of our knowledge, existing theories [39, 45, 61, 36, 44, 58, 49, 62, 21, 17] can only provide implicit formulas in the sense that the approximation error contains an unknown prefactor, or work only for sufficiently large N and L larger than some unknown numbers. For example, [62] estimated an approximation rate $c(d)L^{-2\alpha/d}$ via a narrow and deep ReLU FNN, where $c(d)$ is an unknown number depending on d and L is required to be larger than a sufficiently large unknown number $\mathscr{L}$. For another example, given an approximation error ε, [49] proved the existence of a ReLU FNN with a constant but still unknown number of layers approximating a C^β function within the target error. These works can be divided into two cases: 1) FNNs with varying width

*Submitted to the editors June 13, 2019.

Funding: H. Yang was supported by the startup of the Department of Mathematics at the National University of Singapore and Ministry of Education in Singapore under the grant MOE2018-T2-2-147.

†Department of Mathematics, National University of Singapore (matzuows@nus.edu.sg).

‡Department of Mathematics, National University of Singapore (haizhao@nus.edu.sg).

§Department of Mathematics, National University of Singapore (zhangshijun@u.nus.edu).

and only one hidden layer [18, 23, 37, 60] (visualized by the region in ▭ in Figure 1); 2) FNNs with a fixed width of $\mathcal{O}(d)$ and a varying depth larger than an unknown number $\mathscr{L}$ [40, 62] (represented by the region in ▭ in Figure 1).

As far as we know, the first **quantitative and non-asymptotic** approximation rate of deep ReLU FNNs was obtained in [57]. Specifically, [57] identified an explicit formulas of the approximation rate

$$\begin{cases} 2\nu N^{-2\alpha}, & \text{when } L \geq 2 \text{ and } d = 1, \\ 2(2\sqrt{d})^{\alpha}\nu N^{-2\alpha/d}, & \text{when } L \geq 3 \text{ and } d \geq 2, \end{cases} \tag{1.1}$$

for ReLU FNNs with an arbitrary width $N \in \mathbb{N}^+$ and a fixed depth $L \in \mathbb{N}^+$ to approximate a Hölder continuous function f of order α with a constant ν (visualized in the region shown by ▭ in Figure 1). The approximation rate $\mathcal{O}(N^{-2\alpha/d})$ is tight in terms of N and increasing L cannot improve the approximation rate in N. The success of deep FNNs in a broad range of applications has motivated a well-known conjecture that the depth L has an important role in improving the approximation power of deep FNNs. In particular, a very important question in practice would be, given an arbitrary L and N, what is the explicit formula to characterize the approximation error so as to see whether the network is large enough to meet the accuracy requirement. Due to the highly nonlinear structure of deep FNNs, it is still a challenging open problem to characterize N and L simultaneously in the approximation rate.

To answer the question just above, we establish the first framework that is able to quantify the approximation power of deep ReLU FNNs with arbitrary width N and depth L, achieving the optimal approximation rate, $5\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d})$, for continuous functions $f \in C([0,1]^d)$. Our result is based on new analysis techniques merely based on the structure of FNNs and a modified bit extraction technique inspired by [5], instead of designing FNNs to approximate traditional approximation basis like polynomials and splines as in the existing literature [52, 36, 51, 61, 62, 44, 24, 39, 49, 55, 58]. The approximation rate obtained here admits an explicit formula to compute the prefactor when $\omega_f(\cdot)$ is known. For example, in the case of Hölder continuous functions of order α with a Lipschitz constant ν, $\omega_f(r) \leq \nu r^{\alpha}$ for $r \geq 0$, resulting in the approximation rate $5(8\sqrt{d})^{\alpha}\nu N^{-2\alpha/d}L^{-2\alpha/d}$ as mentioned previously. As a consequence, existing works for the function class $C([0,1]^d)$ are special cases of our result (see Figure 1 for a comparison).

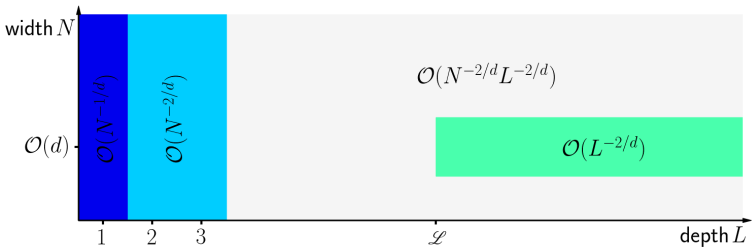

FIG. 1. *A summary of existing and our new results on the approximation rate of ReLU FNNs for continuous functions. Existing results [18, 23, 37, 60, 40, 62, 57] are applicable in the areas in* ▭, ▭, *and* ▭; *our new result is suitable for almost all areas when* $L \geq 2$.

Our main result, Theorem 1.1 below, shows that ReLU FNNs with width $\max\{8d\lfloor N^{1/d}\rfloor + 4d, 12N + 14\}$ and depth $9L + 12$ can approximate f with an approximation rate $5\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d})$, where $\omega_f(\cdot)$ is the modulus of continuity of f defined via

$$\omega_f(r) := \sup\big\{|f(\boldsymbol{x}) - f(\boldsymbol{y})| : \boldsymbol{x}, \boldsymbol{y} \in [0,1]^d,\ |\boldsymbol{x} - \boldsymbol{y}| \leq r\big\}, \quad \text{for any } r \geq 0,$$

where $|\cdot|$ is the absolute value of a scalar or the length of a vector.

THEOREM 1.1. *Let f be a given function in $C([0,1]^d)$. For any arbitrary $L \in \mathbb{N}^+$, $N \in \mathbb{N}^+$, and $\eta > 0$, there exists a ReLU FNN ϕ with width $\max\{8d\lfloor N^{1/d}\rfloor + 4d, 12N + 14\}$ and depth $9L + 12$ such that*

$$\|f - \phi\|_{L^\infty([0,1]^d \setminus \mathcal{H})} \le 5\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d}), \quad \text{and} \quad \|\phi\|_{L^\infty([0,1]^d)} \le |f(0)| + \omega_f(\sqrt{d}),$$

where $\mathcal{H} \subseteq [0,1]^d$ with $\mu(\mathcal{H}) \le \eta$.

Note that ϕ in Theorem 1.1 is uniformly bounded, the measure of $\mathcal{H}$ can be arbitrarily small and is independent of f, N, and L. Hence, the L^p-norm of the approximation error can also be controlled as follows.

COROLLARY 1.2. *Let f be a given function in $C([0,1]^d)$. For arbitrary $L \in \mathbb{N}^+$, $N \in \mathbb{N}^+$, and $p \in [1,\infty)$, there exists a ReLU FNN ϕ with width $\max\{8d\lfloor N^{1/d}\rfloor + 4d, 12N + 14\}$ and depth $9L + 12$ such that*

$$\|f - \phi\|_{L^p([0,1]^d)} \le 5\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d}).$$

When Theorem 1.1 and Corollary 1.2 are applied to a Hölder continuous function f of order α with a Lipschitz constant ν, i.e., $f \in \mathrm{Lip}(\nu, \alpha, d)$, the approximation rate is $5(8\sqrt{d})^\alpha \nu N^{-2\alpha/d}L^{-2\alpha/d}$, because $\omega_f(r) \le \nu r^\alpha$ for any $r \ge 0$. An immediate question following the constructive approximation is how much we can improve the approximation rate. In fact, the approximation rate of $f \in \mathrm{Lip}(\nu, \alpha, d)$ is asymptotically tight based on VC dimension as we shall see later.

In most real applications of neural networks, though the target function f is defined in a high dimensional domain, e.g., $[0,1]^d$, where d could be tens of thousands or even millions, only the approximation error of f in a neighborhood of a $d_{\mathcal{M}}$-dimensional manifold $\mathcal{M}$ with $d_{\mathcal{M}} \ll d$ is concerned. Hence, we extend Theorem 1.1 to the case when the domain of f is localized in an ε-neighborhood of a compact $d_{\mathcal{M}}$-dimensional Riemannian submanifold $\mathcal{M} \subseteq [0,1]^d$ having condition number $1/\tau$, volume V, and geodesic covering regularity $\mathcal{R}$. The ε-neighborhood is defined as

$$\mathcal{M}_\varepsilon \coloneqq \big\{\boldsymbol{x} \in [0,1]^d : \inf\{|\boldsymbol{x} - \boldsymbol{y}| : \boldsymbol{y} \in \mathcal{M}\} \le \varepsilon\big\}, \quad \text{for } \varepsilon \in (0,1). \tag{1.2}$$

Let $d_\delta = \mathcal{O}\Big(\frac{d_{\mathcal{M}} \ln(dV\mathcal{R}\tau^{-1}\delta^{-1})}{\delta^2}\Big) = \mathcal{O}\big(d_{\mathcal{M}}\frac{\ln(d/\delta)}{\delta^2}\big)$ be an integer for any $\delta \in (0,1)$ such that $d_{\mathcal{M}} \le d_\delta \le d$. We show an approximation rate $3\omega_f\big(\frac{4\varepsilon}{1-\delta}\sqrt{\frac{d}{d_\delta}}\big) + 5\omega_f\big(\frac{16d}{(1-\delta)\sqrt{d_\delta}}N^{-2/d_\delta}L^{-2/d_\delta}\big)$ for ReLU FNNs to approximate f in a $L^p([0,1]^d, \mu_\varrho)$-norm, where ϱ is a probability density function (i.e., $\int \varrho(\boldsymbol{x})d\boldsymbol{x} = 1$) supported in $\mathcal{M}_\varepsilon$, and $\mu_\varrho(\cdot)$ is the corresponding measure of ϱ defined via $\mu_g(E) \coloneqq \int_E g(\boldsymbol{x})d\boldsymbol{x}$ for any measurable set $E \subseteq \mathbb{R}^d$. The key ideas of the proof is the application of Theorem 3.1 in [3], which provides a nearly isometric projection $\boldsymbol{A} \in \mathbb{R}^{d_\delta \times d}$ that maps points in $\mathcal{M} \subseteq [0,1]^d$ to a d_δ-dimensional domain with

$$(1-\delta)|\boldsymbol{x}_1 - \boldsymbol{x}_2| \le |\boldsymbol{A}\boldsymbol{x}_1 - \boldsymbol{A}\boldsymbol{x}_2| \le (1+\delta)|\boldsymbol{x}_1 - \boldsymbol{x}_2| \quad \text{for any } \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{M},$$

and the application of Theorem 1.1 in this paper, which constructs the desired ReLU FNN with a size depending on d_δ instead of d to lessen the curse of dimensionality. When δ is closer to 1, d_δ is closer to $d_{\mathcal{M}}$ but the isometric property of the projection is weakened; when δ is closer to 0, the isometric property becomes better but d_δ could be larger than d, in which case we can simply enforce $d_\delta = d$ and choose the identity map as the projection. Hence, $\delta \in (0,1)$ is a parameter to make a balance between isometry and dimension reduction.

THEOREM 1.3. *Let f be a continuous function on $[0,1]^d$ and $\mathcal{M} \subseteq [0,1]^d$ be a compact $d_{\mathcal{M}}$-dimensional Riemannian submanifold. For any $\varepsilon \in (0,1)$, let ϱ be a probability density function supported on $\mathcal{M}_\varepsilon$ defined in (1.2) with $\mu_\varrho(\cdot)$ as its corresponding measure. For any $N \in \mathbb{N}^+$, $L \in \mathbb{N}^+$, $p \in [1,\infty)$, $\delta \in (0,1)$, there exists a ReLU FNN ϕ with width $\max\{8d_\delta\lfloor N^{1/d_\delta}\rfloor + 4d_\delta, 12N + 14\}$ and depth $9L + 12$ such that*

$$\|f - \phi\|_{L^p([0,1]^d, \mu_\varrho)} \le 3\omega_f\big(\tfrac{4\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}}\big) + 5\omega_f\big(\tfrac{16d}{(1-\delta)\sqrt{d_\delta}}N^{-2/d_\delta}L^{-2/d_\delta}\big). \tag{1.3}$$

The approximation rate of deep neural networks for functions defined precisely on low-dimensional smooth manifolds has been studied in [56] for C^2 functions and in [9, 11] for Lipschitz continuous functions. Considering that it might be more reasonable to assume data located in a small neighborhood of low-dimensional smooth manifold in real applications, we introduce the ε-neighborhood of the manifold $\mathcal{M}$ in Theorem 1.3. In general, existing results are again asymptotic and they cannot be applied to estimate the approximation accuracy of a ReLU FNN with arbitrarily given width N and depth L, since there is no explicit formula without unknown constants to specify the exact error bound. For example, [9] provides an approximation rate $c_1 \left(NL\right)^{-c_2/d_\delta}$ with unknown constants (e.g., c_1 and c_2) and requires NL greater than an unknown large number. The demand of an explicit error estimation motivates Theorem 1.3 in this paper. When data are concentrating around $\mathcal{M}$, ε is very small and the dominant term of the approximation error in (1.3) is $5\omega_f\big(\frac{16d}{(1-\delta)\sqrt{d_\delta}} N^{-2/d_\delta} L^{-2/d_\delta}\big)$ implying that the approximation via deep ReLU FNNs can lessen the curse of dimensionality.

The analysis above provides a general guide for selecting the width and depth of ReLU FNNs to approximate continuous functions, especially when the computation is conducted with parallel computing, which is usually the case in real applications [54, 12]. As we shall see later, when the approximation accuracy and the parallel computing efficiency are considered together, very deep FNNs become less attractive than those with $\mathcal{O}(1)$ depth. Besides, the width requirement $\max\big\{8d\lfloor N^{1/d}\rfloor + 4d, 12N + 14\big\}$ in Theorem 1.1 implies that a minimum width of $\mathcal{O}(d)$ is required to maintain a tight approximation rate agreeing with the observation in [39]. In the case of data in an ε-neighborhood of a low-dimensional smooth manifold, the width requirement is reduced to $\max\big\{8d_\delta\lfloor N^{1/d_\delta}\rfloor + 4d_\delta, 12N + 14\big\}$.

The approximation theories in this paper assume that the target function f is fully accessible, making it possible to estimate the approximation error and identify an asymptotically optimal ReLU FNN with a given budget of neurons to minimize the approximation error. In real applications, usually only a limited number of possibly noisy observations of f is available, resulting in a regression problem in statistics. In the latter case, the problem is usually formulated in a stochastic setting with randomly generated noisy observations and the regression error contains mainly two components: bias and variance. The bias is the difference of the expectation of an estimated function and its ground truth f. The approximation theories in this paper play an important role in characterizing the power of neural networks when they are applied to solve regression problems by providing a lower bound of the regression bias.

The rest of this paper is organized as follows. A constructive proof of Theorem 1.1 will be shown in Section 2. An asymptotic analysis will be presented in Section 3 to show the tightness of Theorem 1.1. In Section 4, three aspects of neural networks in practice will be discussed: 1) neural network approximation in a high-dimensional irregular domain; 2) neural network approximation in the case of a low-dimensional data structure; 3) the optimal ReLU FNN in parallel computation. Finally, Section 5 concludes this paper with a short discussion.

2. Constructive Approximation Rate $\mathcal{O}(\omega_f(N^{-2/d}L^{-2/d}))$.

In this section, we prove the quantitative and constructive approximation rate of ReLU FNNs in Theorem 1.1. Notations throughout the proof will be summarized in Section 2.1. The main ideas of the proof are summarized in Section 2.2. The proof of Theorem 1.1 in the one and multi dimensional cases are presented in Section 2.3 and 2.4, respectively. Finally, several auxiliary lemmas required in the proof of Theorem 3.6 are proved in Section 2.5.

2.1. Notations.

Let us summarize all basic notations used in this paper as follows.

- Matrices are denoted by bold uppercase letters. For instance, $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ is a real matrix of size $m \times n$, and $\boldsymbol{A}^T$ denotes the transpose of $\boldsymbol{A}$.
- Vectors are denoted as bold lowercase letters. For example, $\boldsymbol{v} \in \mathbb{R}^n$ is a column vector of size n. Correspondingly, $\boldsymbol{v}(i)$ is the i-th element of $\boldsymbol{v}$. $\boldsymbol{v} = [v_1, \cdots, v_n]^T = \begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix}$ are vectors consisting of numbers $\{v_i\}$ with $\boldsymbol{v}(i) = v_i$.
- Let $\mu(\cdot)$ be the Lebesgue measure.
- The set difference of two sets A and B is denoted by $A\backslash B := \{x : x \in A,\ x \notin B\}$.

- For any $\xi \in \mathbb{R}$, let $\lfloor \xi \rfloor := \max\{i : i \le \xi,\ i \in \mathbb{N}\}$ and $\lceil \xi \rceil := \min\{i : i \ge \xi,\ i \in \mathbb{N}\}$.
- Assume $\boldsymbol{n} \in \mathbb{N}^n$, then $f(\boldsymbol{n}) = \mathcal{O}(g(\boldsymbol{n}))$ means that there exists positive C independent of $\boldsymbol{n}$, f, and g such that $f(\boldsymbol{n}) \le Cg(\boldsymbol{n})$ when all entries of $\boldsymbol{n}$ go to $+\infty$.
- Let $\sigma : \mathbb{R} \to \mathbb{R}$ denote the rectified linear unit (ReLU), i.e. $\sigma(x) = \max\{0, x\}$. With the abuse of notations, we define $\sigma : \mathbb{R}^d \to \mathbb{R}^d$ as $\sigma(\boldsymbol{x}) = \begin{bmatrix} \max\{0, x_1\} \\ \vdots \\ \max\{0, x_d\} \end{bmatrix}$ for any $\boldsymbol{x} = [x_1, \cdots, x_d]^T \in \mathbb{R}^d$.
- We will use NN as a ReLU neural network for short and use Python-type notations to specify a class of NN's, e.g., $\mathrm{NN}(\mathrm{c}_1;\ \mathrm{c}_2;\ \cdots;\ \mathrm{c}_m)$ is a set of ReLU FNNs satisfying m conditions given by $\{\mathrm{c}_i\}_{1 \le i \le m}$, each of which may specify the number of inputs (#input), the total number of nodes in all hidden layers (#node), the number of hidden layers (#layer), the number of total parameters (#parameter), and the width in each hidden layer (widthvec), the maximum width of all hidden layers (maxwidth), etc. For example, if $\phi \in \mathrm{NN}(\#\mathrm{input} = 2;\ \mathrm{widthvec} = [100, 100])$, then ϕ satisfies
  - ϕ maps from $\mathbb{R}^2$ to $\mathbb{R}$.
  - ϕ has two hidden layers and the number of nodes in each hidden layer is 100.
- $[n]^L$ is short for $[n, n, \cdots, n] \in \mathbb{N}^L$. For example,

$$\mathrm{NN}(\#\mathrm{input} = d;\ \mathrm{widthvec} = [100, 100]) = \mathrm{NN}(\#\mathrm{input} = d;\ \mathrm{widthvec} = [100]^2).$$

- For $\phi \in \mathrm{NN}(\#\mathrm{input} = d;\ \mathrm{widthvec} = [N_1, N_2, \cdots, N_L])$, if we define $N_0 = d$ and $N_{L+1} = 1$, then the architecture of ϕ can be briefly described as follows:

$$\boldsymbol{x} = \tilde{\boldsymbol{h}}_0 \xrightarrow{\boldsymbol{W}_1,\ \boldsymbol{b}_1} \boldsymbol{h}_1 \xrightarrow{\sigma} \tilde{\boldsymbol{h}}_1 \cdots \xrightarrow{\boldsymbol{W}_L,\ \boldsymbol{b}_L} \boldsymbol{h}_L \xrightarrow{\sigma} \tilde{\boldsymbol{h}}_L \xrightarrow{\boldsymbol{W}_{L+1},\ \boldsymbol{b}_{L+1}} \phi(\boldsymbol{x}) = \boldsymbol{h}_{L+1},$$

  where $\boldsymbol{W}_i \in \mathbb{R}^{N_i \times N_{i-1}}$ and $\boldsymbol{b}_i \in \mathbb{R}^{N_i}$ are the weight matrix and the bias vector in the i-th linear transform in ϕ, respectively, i.e.,

$$\boldsymbol{h}_i := \boldsymbol{W}_i \tilde{\boldsymbol{h}}_{i-1} + \boldsymbol{b}_i, \quad \text{for } i = 1, \ldots, L+1,$$

  and

$$\tilde{\boldsymbol{h}}_i = \sigma(\boldsymbol{h}_i), \quad \text{for } i = 1, \ldots, L.$$

- The expression, an FNN with width N and depth L, means
  - The maximum width of this FNN for all hidden layers less than or equal to N.
  - The number of hidden layers of this FNN less than or equal to L.
- For $x \in [0, 1)$, suppose its binary representation is $x = \sum_{\ell=1}^{\infty} x_\ell 2^{-\ell}$ with $x_\ell \in \{0, 1\}$, we introduce a special notation $\mathrm{bin}\, 0.x_1 x_2 \cdots x_L$ to denote the L-term binary representation of x, i.e., $\sum_{\ell=1}^{L} x_\ell 2^{-\ell}$.

2.2. Main ideas. We will show that an almost piecewise constant ReLU FNN ϕ is enough to achieve the desired approximation rate in Theorem 1.1. Given an arbitrary $f \in C([0, 1]^d)$, we introduce a piecewise constant function $f_p \approx f$ serving as an intermediate approximant in our construction in the sense that

$$f \approx f_p \approx \phi.$$

The approximation in $f \approx f_p$ is a simple and standard technique in constructive approximation. For example, given arbitrary N and L, uniformly partition $[0, 1]^d$ into $\mathcal{O}(N^2 L^2)$ pieces and define f_p using this partition. Then the approximation error of $f_p \approx f$ scales like $\mathcal{O}(N^{-2/d} L^{-2/d})$. We will address the approximation in $f_p \approx \phi$ with the same error scaling and a limited budget of the FNN size, e.g., $\mathcal{O}(NL)$ neurons, based on the fact that f_p is almost surely a ReLU FNN in $[0, 1]^d \backslash \mathcal{H}$, where $\mathcal{H}$ is a "don't-care" region near the discontinuous locations of f_p with an arbitrarily small Lebesgue measure (see Figure 2 for an illustration). The introduction of the "don't-care" region is to ease the construction of a deep ReLU FNN ϕ, which is a piecewise linear and continuous function, to approximate the discontinuous function f_p by removing the difficulty near discontinuous points, essentially smoothing f_p by restricting the approximation domain in $[0, 1]^d \backslash \mathcal{H}$.

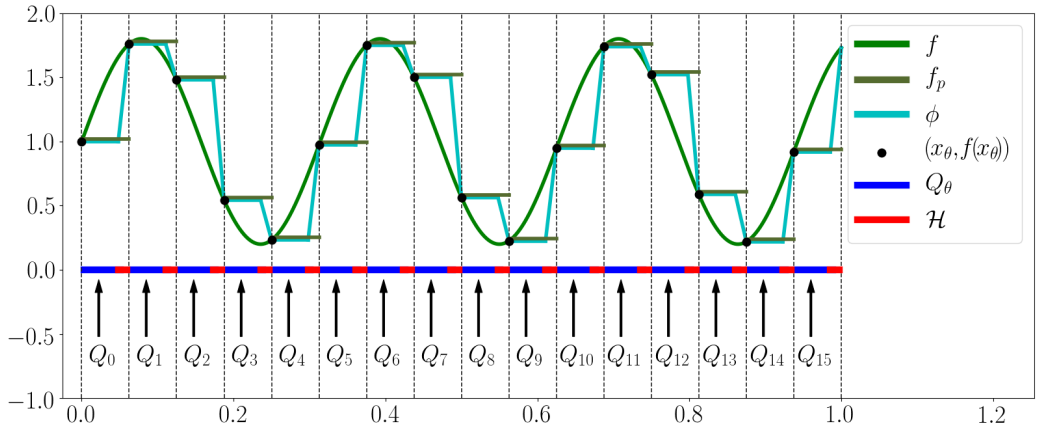

FIG. 2. *An illustration of f, f_p, ϕ, x_θ, Q_θ, and the "don't-care" region $\mathcal{H}$ in the one-dimensional case for $\theta = 0, 1, \cdots, N^2L^2-1$, where $N = 2$ and $L = 2$. f is the target function; f_p is the piecewise constant function approximating f; ϕ is the ReLU FNN approximating f; and x_θ is a representative of Q_θ. The measure of $\mathcal{H}$ can be arbitrarily small as we shall see in the proof of Theorem 1.1.*

In the one-dimensional case, a naive way to construct a ReLU FNN ϕ to approximate f_p is to choose ϕ as a piecewise linear function satisfying: 1) matching f_p in $[0,1]\backslash\mathcal{H}$; 2) having one piece of linear region in each isolated contiguous region of $\mathcal{H}$ (see ϕ and $\mathcal{H}$ in Figure 2 for an illustration). Such a ϕ has $\mathcal{O}(N^2L^2)$ pieces of linear regions. It can be represented as a ReLU FNN with one hidden layer and $\mathcal{O}(N^2L^2)$ neurons, which is far beyond our budget, e.g., $\mathcal{O}(NL)$ neurons. This motivates our design of a deep ReLU FNN via function compositions for the limited budget.

Let us use the one-dimensional case and Figure 2 to illustrate the construction of $\phi(x) \approx f_p(x)$ for $x \in [0,1]\backslash\mathcal{H}$ as a warm-up. First of all, we simplify the function approximation $\phi(x) \approx f_p(x)$ for $x \in [0,1]\backslash\mathcal{H}$ in the L^∞-norm to a regression problem with N^2L^2 samples $\{x_\theta, f_p(x_\theta)\}_{\theta\in\{0,1,\ldots,N^2L^2-1\}}$, since in the one-dimensional case f_p has N^2L^2 pieces $\{Q_\theta\}_\theta$ and constant values $\{f_p(x_\theta)\}_\theta$, where x_θ is a point representative in Q_θ, e.g., an end point of Q_θ as in Figure 2, and $\{Q_\theta\}_\theta$ are the intervals separated by $\mathcal{H}$.

We design a ReLU FNN $\phi = \tilde{\phi}_2 \circ \phi_1$ to approximate f_p, where ϕ_1 and $\tilde{\phi}_2$ are two ReLU FNNs such that $\phi_1(x) = x_\theta$ for $x \in Q_\theta$ and $f_p(x_\theta) = \tilde{\phi}_2(x_\theta) \quad \text{for} \quad \theta \in \{0, 1, \ldots, N^2L^2 - 1\}$. Then $f_p(x) \approx \phi(x) = \tilde{\phi}_2 \circ \phi_1(x)$ for any $x \in [0,1]\backslash\mathcal{H}$. In fact, for the convenience of indexing, we construct ϕ_1 and $\tilde{\phi}_2$ such that $\phi_1(x) = \theta$ for $x \in Q_\theta$ and $f_p(x_\theta) = \tilde{\phi}_2(\theta) \quad \text{for} \quad \theta \in \{0, 1, \ldots, N^2L^2 - 1\}$. See Figure 3 (a) and (b) for an example of ϕ_1. The simplification reduces the difficulty of controlling the approximation error for all points in $[0,1]\backslash\mathcal{H}$ to a simpler task of controlling the regression error at N^2L^2 points. The most difficult part of the proof of Theorem 1.1 is to construct such two ReLU FNNs ϕ_1 and $\tilde{\phi}_2$ using $\mathcal{O}(NL)$ neurons.

Similarly in the d-dimensional case, $[0,1]^d$ is divided into a union of important regions $\{Q_{\boldsymbol{\theta}}\}$ and a "don't-care" region $\mathcal{H}$, where each $Q_{\boldsymbol{\theta}}$ is associated with a representative $\boldsymbol{x}_{\boldsymbol{\theta}} \in Q_{\boldsymbol{\theta}}$ such that $f(\boldsymbol{x}_{\boldsymbol{\theta}}) = f_p(\boldsymbol{x}_{\boldsymbol{\theta}})$ for each index vector $\boldsymbol{\theta} \in \{0, 1, \ldots, J-1\}^d$, where $J = \mathcal{O}(N^{2/d}L^{2/d})$ is the partition number per dimension (see Figure 3 (c) for an example when $d = 2$). We design a ReLU FNN $\phi = \tilde{\phi}_2 \circ \boldsymbol{\phi}_1$ to approximate f_p, where $\boldsymbol{\phi}_1$ and $\tilde{\phi}_2$ are two ReLU FNNs such that $\boldsymbol{\phi}_1(\boldsymbol{x}) = \boldsymbol{\theta}$ for $\boldsymbol{x} \in Q_{\boldsymbol{\theta}}$ and $f_p(\boldsymbol{x}_{\boldsymbol{\theta}}) = \tilde{\phi}_2(\boldsymbol{\theta}) \quad \text{for} \quad \boldsymbol{\theta} \in \{0, 1, \ldots, J-1\}^d$. Then $\phi(\boldsymbol{x}) \approx f_p(\boldsymbol{x})$ for $\boldsymbol{x} \in [0,1]^d\backslash\mathcal{H}$. Here, $\boldsymbol{\phi}_1(\boldsymbol{x})$ can be constructed via $\boldsymbol{\phi}_1(\boldsymbol{x}) = [\phi_1(x_1), \phi_1(x_2), \cdots, \phi_1(x_d)]^T$, where each one-dimensional ReLU FNN ϕ_1 follows the same construction as in the one-dimensional case.

However, constructing a d-dimensional ReLU FNN $\tilde{\phi}_2$ for the d-dimensional regression problem above is not straightforward, which motivates us to reduce the d-dimensional regression problem to a one-dimensional regression problem via an injective projection map ϕ_2 in a form of ReLU FNN such that it maps $\boldsymbol{\theta} \in \{0, 1, \ldots, J-1\}^d$ to $\mathbb{N}$. Then we construct a ReLU FNN ϕ_3 such that $\tilde{\phi}_2(\boldsymbol{\theta}) = \phi_3 \circ \phi_2(\boldsymbol{\theta}) \quad \text{for} \quad \boldsymbol{\theta} \in \{0, 1, \ldots, J-1\}^d$. The

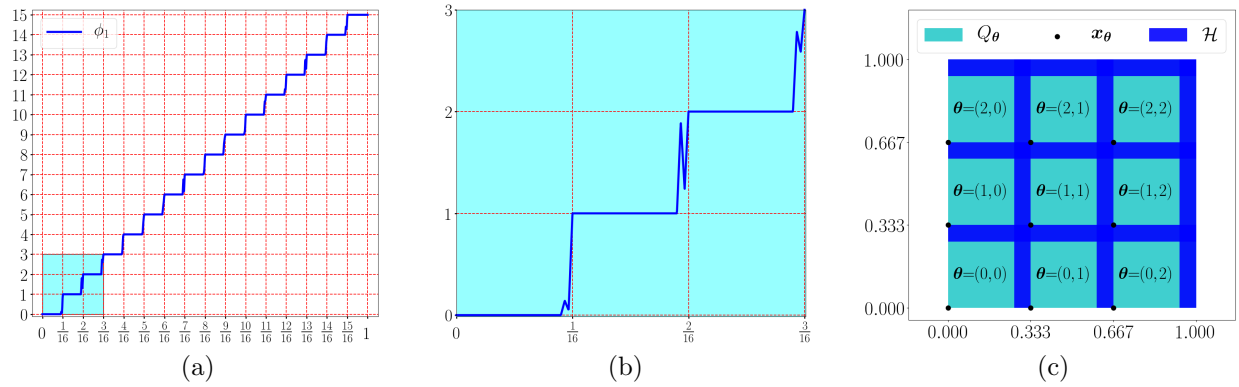

FIG. 3. *(a) An illustration of ϕ_1 when $d = 1$, $N = 2$, and $L = 2$. (b) A zoomed-in image of the cyan region in (a). In the "don't-care" region $\mathcal{H}$, e.g., around $\frac{1}{16}$ and $\frac{2}{16}$, ϕ_1 may oscillate and it is difficult to control its behavior. (c) An illustration of $Q_{\boldsymbol{\theta}}$, $\boldsymbol{x}_{\boldsymbol{\theta}}$, and the "don't-care" region $\mathcal{H}$ when $d = 2$ and the partition number per dimension $J = 3$.*

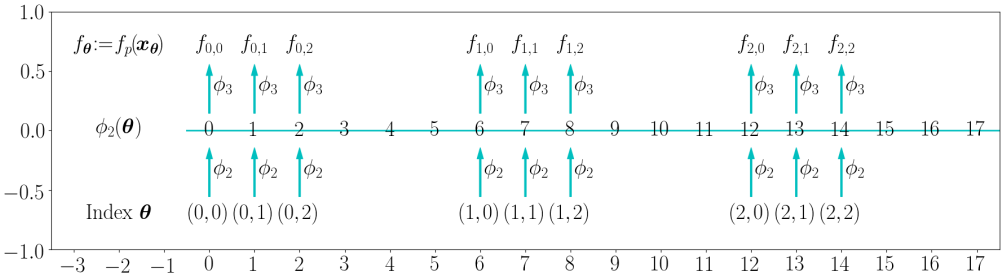

FIG. 4. *An illustration of ϕ_2 and ϕ_3 when $d = 2$ and the partition number per dimension $J = 3$. The indices $\{\boldsymbol{\theta}\}$ are corresponding to those in Figure 3 (c). For example, given $\boldsymbol{\theta} = (0,0)$ and $\boldsymbol{x}$ in $Q_{\boldsymbol{\theta}}$ in Figure 3 (c), $\boldsymbol{\phi}_1$ maps $\boldsymbol{x}$ to $\boldsymbol{\theta}$, which is mapped to 0 by ϕ_2 as in this figure. Next, 0 will be mapped to $f_{\boldsymbol{\theta}} := f_p(\boldsymbol{x}_{\boldsymbol{\theta}})$ by ϕ_3 finally.*

construction of ϕ_3 is the same as that of $\tilde{\phi}_2$ in the one-dimensional case since they are ReLU FNNs mapping $\mathcal{O}(N^2L^2)$ integers to given function values. Hence, we can also use the notation $\phi_3 \circ \phi_2(\boldsymbol{\theta})$ when $d = 1$ via setting ϕ_2 as an identity map. The computation flow of $\boldsymbol{\phi}_1$, ϕ_2, and ϕ_3 is illustrated in Figure 4, and the overall structure of the ReLU FNN ϕ constructed in a form of $\phi_3 \circ \phi_2 \circ \boldsymbol{\phi}_1$ is visualized in Figure 5.

Finally, we discuss how to construct $\boldsymbol{\phi}_1$, ϕ_2, and ϕ_3 using deep ReLU FNNs with width $\mathcal{O}(N)$ and depth $\mathcal{O}(L)$ using several propositions as we shall prove in Section 2.5 later. The construction of $\boldsymbol{\phi}_1$ is based on Proposition 2.1 and 2.2 below. Directly constructing a deep FNN achieving an approximation goal is challenging, while designing a shallow FNN with one hidden layer for the same approximation task is simpler. This motivates the introduction of Proposition 2.1 and 2.2 to show how to construct a deep FNN to approximate a shallow FNN. As shown in Figure 6, suppose a shallow FNN (shown in Figure 6 (left)) is able to fit given samples in Proposition 2.1, the application of Proposition 2.1 leads to an FNN with two hidden layers (shown in Figure 6 (middle)) approximating this shallow FNN, while the application of Proposition 2.2 results in a deep FNN (shown in Figure 6 (right)) approximating the two-hidden-layer FNN. Hence, the resulting deep FNN can also fit the same given samples as the shallow FNN does. This idea as visualized in Figure 6 will be repeatedly applied in the proof of Theorem 1.1.

PROPOSITION 2.1. *For any $N_1, N_2 \in \mathbb{N}^+$, given $N_1(N_2+1)+1$ samples $(x_i, y_i) \in \mathbb{R}^2$ with $x_0 < x_1 < \cdots < x_{N_1(N_2+1)}$ and $y_i \geq 0$ for $i = 0, 1, \cdots, N_1(N_2+1)$, there exists $\phi \in \mathrm{NN}(\#\text{input} = 1; \text{widthvec} = [2N_1, 2N_2+1])$ satisfying the following conditions.*

1. *$\phi(x_i) = y_i$ for $i = 0, 1, \cdots, N_1(N_2+1)$;*

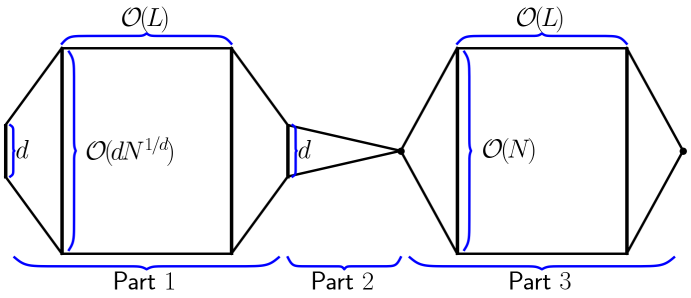

FIG. 5. *The structure of the target FNN $\phi = \phi_3 \circ \phi_2 \circ \phi_1$. The first part of ϕ is ϕ_1, an FNN with width $\mathcal{O}(dN^{1/d})$ and depth $\mathcal{O}(L)$ constructed via Proposition 2.1 and 2.2; the second part of ϕ is ϕ_2, an FNN with an input layer, an output layer, and no hidden layer; the third part of ϕ is ϕ_3, an FNN with width $\mathcal{O}(N)$ and depth $\mathcal{O}(L)$ constructed via Proposition 2.3.*

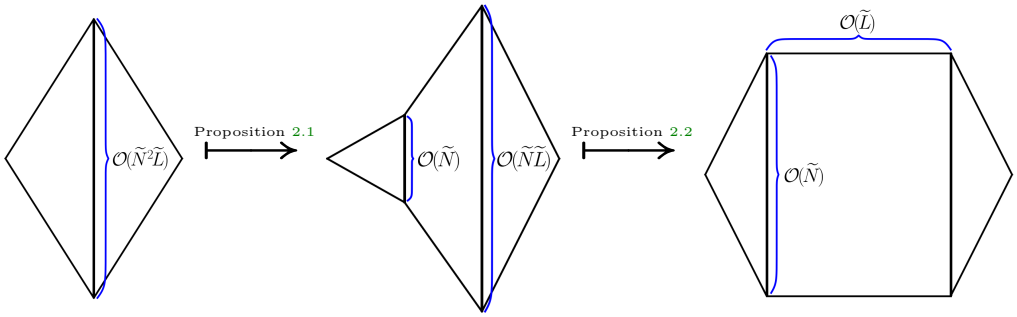

FIG. 6. *Given $\mathcal{O}(\widetilde{N}^2\widetilde{L})$ samples, it is simple to construct a ReLU FNN with one hidden layer of size $\mathcal{O}(\widetilde{N}^2\widetilde{L})$ fitting these samples. The application of Proposition 2.1 with $N_1 = \mathcal{O}(\widetilde{N})$ and $N_2 = \mathcal{O}(\widetilde{N}\widetilde{L})$ gives a ReLU FNN with two hidden layers fitting the same samples as well. Finally, the application of Proposition 2.2 with $N = \widetilde{N}$ and $L = \widetilde{L}$ shows the existence of a deep FNN equivalent to the two-hidden-layer FNN.*

2. *ϕ is linear on each interval $[x_{i-1}, x_i]$ for $i \notin \{(N_2+1)j : j = 1, 2, \cdots, N_1\}$.*

PROPOSITION 2.2. *Given any $N, L \in \mathbb{N}^+$, for arbitrary $\phi_1 \in \mathrm{NN}(\#\text{input} = 1; \text{widthvec} = [N, NL])$, there exists $\phi_2 \in \mathrm{NN}(\#\text{input} = 1; \text{maxwidth} \le 2N + 4; \#\text{layer} \le L + 2)$ such that $\phi_1(x) = \phi_2(x)$ for any $x \in \mathbb{R}$.*

Afterwards, we will use the notation $\mathrm{NN}(\#\text{input} = 1; \text{widthvec} = [N, NL]) \subseteq \mathrm{NN}(\#\text{input} = 1; \text{maxwidth} \le 2N + 4; \#\text{layer} \le L + 2)$ in the sense of Proposition 2.1.

The construction of ϕ_2 is inspired by the J-ary representation, i.e., $\phi_2(\boldsymbol{\theta}) = 2J^d\big(\frac{\theta_d}{2J^d} + \sum_{i=1}^{d-1} \frac{\theta_i}{J^i}\big)$ for $\boldsymbol{\theta} \in \{0, 1, \cdots, J-1\}^d$, which can be implemented easily via a one-layer ReLU FNN as we shall see in the proof of Theorem 1.1 later.

The construction of ϕ_3 is a direct result of Proposition 2.3 below, the proof of which relies on both the idea illustrated in Figure 6 and the bit extraction technique in [5].

PROPOSITION 2.3. *Given any $\varepsilon > 0$, arbitrary $N, L, K \in \mathbb{N}^+$ with $K \le N^2L^2$, a sample set $\{y_k \ge 0 : k = 0, 1, \cdots, K-1\}$ with $|y_k - y_{k-1}| \le \varepsilon$, for $k = 1, 2, \cdots, K-1$, there exists $\phi \in \mathrm{NN}(\#\text{input} = 1; \text{maxwidth} \le 12N + 14; \#\text{layer} \le 4L + 10)$ such that*

1. *$|\phi(k) - y_k| \le \varepsilon$, for $k = 0, 1, \cdots, K-1$;*
2. *$0 \le \phi(t) \le \max\{y_k : k = 0, 1, \cdots, K-1\}$, for any $t \in \mathbb{R}$.*

Due to the highly nonlinear nature of function compositions, though it is possible to construct ϕ as a deep ReLU FNN approximating f_p uniformly well in $[0,1]^d \backslash \mathcal{H}$ with the desired approximation rate using the above ideas, the configuration of ϕ might be complicated in $\mathcal{H}$ and we can only bound the L^∞-norm of ϕ in $\mathcal{H}$ with a constant independent of N and L. In fact, the behavior of $\phi(\boldsymbol{x})$ when $\boldsymbol{x} \in \mathcal{H}$ may oscillate, e.g., due to

the oscillation of ϕ_1 as shown in Figure 3 (b). Fortunately, we would like to emphasize that our construction of such a ϕ is valid for $\mathcal{H}$ with an arbitrarily small measure. Hence, it is enough to care about the L^∞-norm of ϕ in the "don't-care" region $\mathcal{H}$, since the approximation is essentially valid in $[0,1]^d$ in the L^∞ sense and completely valid in the L^p sense for $p \in [1,\infty)$.

With the above propositions ready, let us prove Theorem 1.1 in the case of $d = 1$ in Section 2.3 and the case of $d \geq 2$ in Section 2.4. We further assume that $\omega_f(r) > 0$ for any $r > 0$, excluding a simple case when f is a constant function.

2.3. Proof of Theorem 1.1 for The Case $d = 1$. The key idea for the proof of Theorem 1.1 is to construct an almost piecewise constant ReLU FNN to approximate f within a budget of $\mathcal{O}(NL)$ neurons to achieve the desired approximation power. Note that $|f(x)-f(0)| \leq \omega_f(1)$ for $x \in [0,1]$. Set $\bar{f} = f - f(0) + \omega_f(1)$, then $0 \leq \bar{f}(x) \leq 2\omega_f(1)$ for $x \in [0,1]$. Let $M = N^2L$ and $\delta > 0$ be a small number such that

$$ML\delta = N^2L^2\delta \leq \min\{1/2, \eta\}. \tag{2.1}$$

The proof can be divided into four steps as follows:

1. Divide $[0,1]$ into a union of sub-intervals $\{Q_\theta : \theta = 0, 1, \cdots, ML-1\}$ and a "don't-care" region $\mathcal{H}$;
2. Construct a sub-FNN ϕ_1 to output θ given $x \in Q_\theta$, i.e., $\phi_1(x) = \theta$;
3. Construct a sub-FNN ϕ_2 mapping the index θ approximately to $\bar{f}(\frac{\theta}{ML}) = f(\frac{\theta}{ML}) - f(0) + \omega_f(1)$;
4. Construct the final target FNN $\phi = \phi_2 \circ \phi_1$.

Step 1: Divide $[0,1]$ into a union of sub-intervals $\{Q_\theta\}_\theta$ and a "don't-care" region $\mathcal{H}$.

Define

$$\mathcal{H} := \cup_{\theta=1}^{ML} [\tfrac{\theta}{ML} - \delta, \tfrac{\theta}{ML}], \tag{2.2}$$

and

$$Q_\theta := [\tfrac{\theta}{ML}, \tfrac{\theta+1}{ML} - \delta], \quad \text{for } \theta = 0, 1, \cdots, ML-1.$$

Apparently, $\mu(\mathcal{H}) \leq ML\delta$ and $[0,1] = \left(\cup_{\theta=0}^{ML-1} Q_\theta\right) \cup \mathcal{H}$. See Figure 2 for an illustration of $\{Q_\theta\}_\theta$ and the "don't-care" region $\mathcal{H}$.

Step 2: Construct a sub-FNN ϕ_1 mapping $x \in Q_\theta$ to θ.

There are $ML = N^2L^2$ intervals $\{Q_\theta\}$. It is difficult to directly construct a ReLU FNN with at most $\mathcal{O}(NL)$ neurons mapping $x \in Q_\theta$ to θ for $\theta = 0, 1, \cdots, N^2L^2 - 1$. However, there exists a ReLU FNN with at most $\mathcal{O}(NL)$ neurons mapping $x \in Q_\theta$ to θ for N^2L θ's by Proposition 2.1. Note that the one-dimensional index θ is equivalent to a two dimensional index pair (m, ℓ) via $\theta = mL + \ell$. Hence, we can apply Proposition 2.1 to construct FNNs ψ_1 and ψ_2 with at most $\mathcal{O}(NL)$ neurons such that $\psi_1(x) = m$ and $\psi_2(x) = \ell$, and construct the desired FNN ϕ_1 via the linear combination of ψ_1 and ψ_2 as follows.

For the sample set $\{(\frac{m}{M}, m) : m = 0, 1, \cdots, M\} \cup \{(\frac{m+1}{M} - \delta, m) : m = 0, 1, \cdots, M-1\} \subseteq \mathbb{R}^2$, whose cardinality is $2M + 1 = N \cdot \big((2NL-1)+1\big) + 1$, by Proposition 2.1, there exist $\psi_1 \in \mathrm{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 2(2NL-1)+1]) = \mathrm{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 4NL-1])$ such that

- $\psi_1(1) = M$ and $\psi_1(\frac{m}{M}) = \psi_1(\frac{m+1}{M} - \delta) = m$ for $m = 0, 1, \cdots, M-1$;
- ψ_1 is linear on each interval $[\frac{m}{M}, \frac{m+1}{M} - \delta]$ for $m = 0, 1, \cdots, M-1$.

Then

$$\psi_1(x) = m, \quad \text{if } x \in Q_\theta \text{ and } \theta = mL + \ell, \quad \text{for } m = 0, 1, \cdots, M-1,\ \ell = 0, 1, \cdots, L-1. \tag{2.3}$$

For the sample set $\{(\frac{\ell}{ML}, \ell) : \ell = 0, 1, \cdots, L\} \cup \{(\frac{\ell+1}{ML} - \delta, \ell) : \ell = 0, 1, \cdots, L-1\} \subseteq \mathbb{R}^2$, whose cardinality is $2L + 1 = 1 \cdot \big((2L-1)+1\big) + 1$, by Proposition 2.1, there exists $\psi_2 \in \mathrm{NN}(\#\text{input} = 1;\ \text{widthvec} = [2, 2(2L-1)+1]) = \mathrm{NN}(\#\text{input} = 1;\ \text{widthvec} = [2, 4L-1])$ such that

- $\psi_2(\frac{1}{M}) = L$ and $\psi_2(\frac{\ell}{ML}) = \psi_2(\frac{\ell+1}{ML} - \delta) = \ell$ for $\ell = 0, 1, \cdots, L-1$;
- ψ_2 is linear on each interval $[\frac{\ell}{ML}, \frac{\ell+1}{ML} - \delta]$ for $\ell = 0, 1, \cdots, L-1$.

Then

$$\psi_2\big(x - \tfrac{1}{M}\psi_1(x)\big) = \psi_2(x - \tfrac{m}{M}) = \ell, \quad \text{if } x \in Q_\theta \text{ and } \theta = mL + \ell, \quad \text{for } m = 0, 1, \cdots, M-1,\ \ell = 0, 1, \cdots, L-1. \tag{2.4}$$

Define

$$\phi_1(x) \coloneqq L\psi_1(x) + \psi_2\big(x - \tfrac{1}{M}\psi_1(x)\big), \quad \text{for any } x \in \mathbb{R}.$$

By (2.3) and (2.4), if $x \in Q_\theta$ and $\theta = mL + \ell$ for $m = 0, 1, \cdots, M-1,\ \ell = 0, 1, \cdots, L-1$, we have

$$\phi_1(x) = L\psi_1(x) + \psi_2\big(x - \tfrac{1}{M}\psi_1(x)\big) = Lm + \psi_2(x - \tfrac{m}{M}) = Lm + \ell = \theta.$$

By Proposition 2.2, $\psi_1 \in \mathrm{NN}(\#\text{input} = 1; \text{widthvec} = [2N, 4NL-1]) \subseteq \mathrm{NN}(\#\text{input}; \text{maxwidth} \le 8N + 4; \#\text{layer} \le L + 2)$ and $\psi_2 \in \mathrm{NN}(\#\text{input} = 1; \text{widthvec} = [2, 4L-1]) \subseteq \mathrm{NN}(\#\text{input}; \text{maxwidth} \le 12; \#\text{layer} \le L + 2)$, then $\phi_1 \in \mathrm{NN}(\#\text{input} = 1; \text{maxwidth} \le 4N + 10; \#\text{layer} \le 2L + 4)$.

Step 3: Construct a sub-FNN ϕ_2 mapping the index θ approximately to $\bar{f}(\frac{\theta}{ML})$.

Define

$$y_\theta \coloneqq \bar{f}(\tfrac{\theta}{ML}) \ge 0, \quad \text{for } \theta = 0, 1, \cdots, ML-1,$$

then $|y_\theta - y_{\theta-1}| \le \omega_f(\frac{1}{ML})$. By Proposition 2.3, there exists $\phi_2 \in \mathrm{NN}(\#\text{input} = 2; \text{maxwidth} \le 12N+14; \#\text{layer} \le 4L + 10)$ such that

$$|\phi_2(\theta) - y_\theta| \le \omega_f(\tfrac{1}{ML}), \quad \text{for } \theta = 0, 1, \cdots, ML-1, \tag{2.5}$$

and

$$0 \le \phi_2(t) \le \max\{y_\theta : \theta = 0, 1, \cdots, ML-1\} \le 2\omega_f(1), \quad \text{for any } t \in \mathbb{R}. \tag{2.6}$$

Step 4: Construct the final target FNN ϕ using ϕ_1 and ϕ_2.

Define

$$\bar{\phi} \coloneqq \phi_2 \circ \phi_1 \quad \text{and} \quad \phi \coloneqq \bar{\phi} + f(0) - \omega_f(1).$$

Then $\phi = \bar{\phi} + f(0) - \omega_f(1) \in \mathrm{NN}(\#\text{input} = 1; \text{maxwidth} \le 12N + 14; \#\text{layer} \le 6L + 14)$ as desired. It suffices to estimate the approximation rate. By Equation (2.6), it holds that

$$0 \le \bar{\phi}(x) = \phi_2 \circ \phi_1(x) \le 2\omega_f(1), \quad \text{for any } x \in [0, 1]. \tag{2.7}$$

By Equation (2.3), (2.4), and (2.5), for $x \in Q_\theta$, $\theta = 0, 1, \cdots, ML-1$, we have

$$|\bar{f}(x) - \bar{\phi}(x)| = \big|\bar{f}(x) - \phi_2 \circ \phi_1(x)\big| \le |\bar{f}(x) - y_\theta| + |y_\theta - \phi_2(\theta)| = |\bar{f}(x) - \bar{f}(\tfrac{\theta}{ML})| + \omega_f(\tfrac{1}{ML}) \le 2\omega_f(\tfrac{1}{ML}).$$

It follows that

$$|\phi(x) - f(x)| = |\bar{f}(x) - \bar{\phi}(x)| \le 2\omega_f(\tfrac{1}{ML}) = 2\omega_f(N^{-2}L^{-2}), \quad \text{for any } x \in [0, 1] \backslash \mathcal{H}. \tag{2.8}$$

By (2.1), (2.2), and (2.7), we get $\mu(\mathcal{H}) \le ML\delta \le \eta$ and $\|\phi\|_{L^\infty([0,1])} \le |f(0)| + \omega_f(1)$. So we finish the proof for the case $d = 1$.

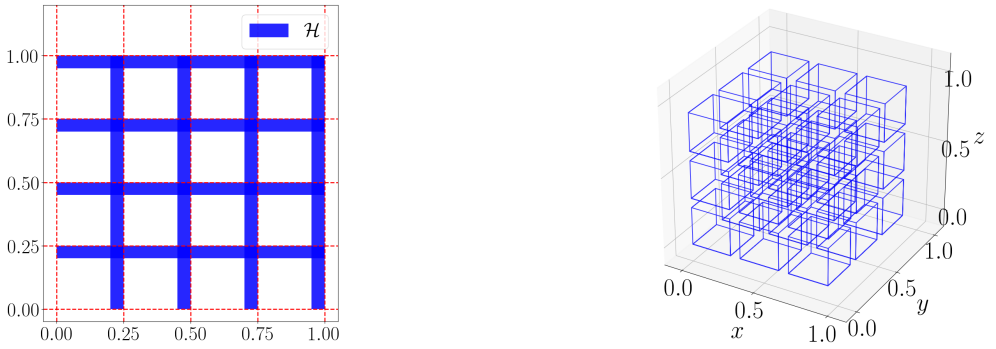

FIG. 7. *(a) An illustration of the "don't-care" region $\mathcal{H}$ for $J = 4$ and $d = 2$. (b) An illustration of $Q_{\boldsymbol{\theta}}$ sub-cubes depicted in blue for $\boldsymbol{\theta} \in \{0, 1, \cdots, J-1\}^d$, where $J = 3$ and $d = 3$.*

2.4. Proof of Theorem 1.1 for The Case $d \geq 2$. We also need to construct an almost piecewise constant ReLU FNN with $\mathcal{O}(NL)$ neurons to approximate f in the d-dimensional case. The main difficulty is to design a ReLU FNN to reduce the d-dimensional approximation problem to a one-dimensional approximation problem as introduced in Section 2.2. It is clear that $|f(\boldsymbol{x}) - f(0)| \leq \omega_f(\sqrt{d})$ for any $\boldsymbol{x} \in [0,1]^d$. Define $\bar{f} = f - f(0) + \omega_f(\sqrt{d})$, then $0 \leq \bar{f}(\boldsymbol{x}) \leq 2\omega_f(\sqrt{d})$ for any $\boldsymbol{x} \in [0,1]^d$. Let $M = N^2L$, $J = \lfloor N^{1/d} \rfloor^2 \lfloor L^{2/d} \rfloor$, and $\delta = \delta(N, L, d, f) > 0$ be a sufficiently small number satisfying

$$Jd\delta = \lfloor N^{1/d} \rfloor^2 \lfloor L^{2/d} \rfloor d\delta \leq \min\{1, \eta\}. \tag{2.9}$$

The proof can be divided into six steps as follows:

1. Divide $[0,1]^d$ into a union of sub-cubes $\{Q_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta} \in \{0,1,\cdots,J-1\}^d}$ and a "don't-care" region $\mathcal{H}$;
2. Construct a sub-FNN $\boldsymbol{\phi}_1$ mapping $\boldsymbol{x}$ to the d-dimensional index $\boldsymbol{\theta}$ if $\boldsymbol{x} \in Q_{\boldsymbol{\theta}}$;
3. Construct a sub-FNN ϕ_2 mapping the index set $\{0, 1, \cdots, J-1\}^d$ to an auxiliary set $\mathcal{A}_1 \subseteq \left\{\frac{k}{2J^d} : k = 0, 1, \cdots, 2J^d\right\}$;
4. Find a continuous piecewise linear function g such that $\bar{f}$ and $g \circ \phi_2 \circ \boldsymbol{\phi}_1$ have the same value at the elements of $\{0, 1, \cdots, J-1\}^d/J$, i.e. $\bar{f} \approx g \circ \phi_2 \circ \boldsymbol{\phi}_1$;
5. Construct a sub-FNN ϕ_3 mapping $\mathcal{A}_1$ to $\left\{\bar{f}(\boldsymbol{x}) : \boldsymbol{x} \in \{0, 1, \cdots, J-1\}^d/J\right\}$ such that $g \approx \phi_3$ on $\mathcal{A}_1$;
6. Construct the final target FNN $\phi = \phi_3 \circ \phi_2 \circ \boldsymbol{\phi}_1 + f(0) - \omega_f(\sqrt{d}) \approx \bar{f} + f(0) - \omega_f(\sqrt{d}) = f$.

Step 1: Divide $[0,1]^d$ into a union of sub-cubes $\{Q_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta} \in \{0,1,\cdots,J-1\}^d}$ and a "don't-care" region $\mathcal{H}$.

Define

$$\mathcal{H} = \cup_{j=1}^d \left\{\boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in [0,1]^d : x_j \in \cup_{i=1}^J [\tfrac{i}{J} - \delta, \tfrac{i}{J}]\right\},$$

which is the so called "don't-care" region and it separates the d-dimensional cube into J^d "important" sub-cubes. To index these d-dimensional smaller sub-cubes, define

$$Q_{\boldsymbol{\theta}} = \{\boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in [0,1]^d : x_i \in [\tfrac{\theta_i}{J}, \tfrac{\theta_i+1}{J} - \delta],\ i = 1, 2, \cdots, d\}$$

for each d-dimensional index vector $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_d]^T \in \{0, 1, \cdots, J-1\}^d$. Apparently, $\mu(\mathcal{H}) \leq Jd\delta$, $[0,1]^d = \cup_{\boldsymbol{\theta} \in \{0,1,\cdots,J-1\}^d} Q_{\boldsymbol{\theta}} \cup \mathcal{H}$, and the intersection of $\mathcal{H}$ and $Q_{\boldsymbol{\theta}}$ belongs to the boundary of $Q_{\boldsymbol{\theta}}$ (see Figure 7 for illustrations).

Step 2: Construct a sub-FNN $\boldsymbol{\phi}_1$ mapping $\boldsymbol{x}$ to the d-dimensional index $\boldsymbol{\theta}$ if $\boldsymbol{x} \in Q_{\boldsymbol{\theta}}$.

For the sample set $\{(\frac{j}{J}, j) : j = 0, 1, \cdots, J\} \cup \{(\frac{j+1}{J} - \delta, j) : j = 0, 1, \cdots, J-1\} \subseteq \mathbb{R}^2$, whose cardinality is $2J+1 = \lfloor N^{1/d} \rfloor \cdot \big((2\lfloor N^{1/d} \rfloor \lfloor L^{2/d} \rfloor - 1) + 1\big) + 1$, by Proposition 2.1, there exists $\phi_1 \in \mathrm{NN}(\#\text{input} = 1; \text{widthvec} = [2\lfloor N^{1/d} \rfloor, 2(2\lfloor N^{1/d} \rfloor \lfloor L^{2/d} \rfloor - 1) + 1]) = \mathrm{NN}(\#\text{input} = 1; \text{widthvec} = [2\lfloor N^{1/d} \rfloor, 4\lfloor N^{1/d} \rfloor \lfloor L^{2/d} \rfloor - 1])$ such that

- $\phi_1(1) = J$, and $\phi_1(\frac{j}{J}) = \phi_1(\frac{j+1}{J} - \delta) = j$ for $j = 0, 1, \cdots, J-1$;
- ϕ_1 is linear on each interval $[\frac{j}{J}, \frac{j+1}{J} - \delta]$ for $j = 0, 1, \cdots, J-1$.

Then $\boldsymbol{\theta} = [\phi_1(x_1), \phi_1(x_2), \cdots, \phi_1(x_d)]^T$ for any $\boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in Q_{\boldsymbol{\theta}}$. Define

$$\boldsymbol{\phi}_1(\boldsymbol{x}) \coloneqq [\phi_1(x_1), \phi_1(x_2), \cdots, \phi_1(x_d)]^T, \quad \text{for any } \boldsymbol{x} \in \mathbb{R}^d.$$

Hence, $\boldsymbol{\phi}_1(\boldsymbol{x}) = \boldsymbol{\theta}$ if $\boldsymbol{x} \in Q_{\boldsymbol{\theta}}$ for $\boldsymbol{\theta} \in \{0, 1, \cdots, J-1\}^d$.

Step 3: Construct a sub-FNN ϕ_2 mapping the index set $\{0, 1, \cdots, J-1\}^d$ to an auxiliary set $\mathcal{A}_1 \subseteq \big\{\frac{k}{2J^d} : k = 0, 1, \cdots, 2J^d\big\}$.

Inspired by the binary representation, we define

$$\phi_2(\boldsymbol{\theta}) = \phi_2(\theta_1, \theta_2, \cdots, \theta_d) \coloneqq \tfrac{\theta_d}{2J^d} + \sum_{i=1}^{d-1} \tfrac{\theta_i}{J^i}, \tag{2.10}$$

then ϕ_2 is an FNN mapping the index set $\{0, 1, \cdots, J-1\}^d$ to

$$\mathcal{A}_1 \coloneqq \Big\{ \tfrac{\theta_d}{2J^d} + \sum_{i=1}^{d-1} \tfrac{\theta_i}{J^i} : \boldsymbol{\theta} \in \{0, 1, \cdots, J-1\}^d \Big\} \subseteq \big\{ \tfrac{k}{2J^d} : k = 0, 1, \cdots, 2J^d \big\}.$$

Step 4: Construct a continuous piecewise linear function g such that $\bar{f}$ and $g \circ \phi_2 \circ \boldsymbol{\phi}_1$ have the same value at the elements of $\{0, 1, \cdots, J-1\}^d / J$.

Define another auxiliary set

$$\mathcal{A}_2 \coloneqq \Big\{ \tfrac{1}{2J^{d-1}} + \tfrac{\theta_d}{2J^d} + \sum_{i=1}^{d-1} \tfrac{\theta_i}{J^i} : \boldsymbol{\theta} \in \{0, 1, \cdots, J-1\}^d \Big\},$$

then $\mathcal{A}_1 \cup \mathcal{A}_2 \cup \{1\} = \{\frac{k}{2J^d} : k = 0, 1, \cdots, 2J^d\}$.

Let $g : [0, 1] \to \mathbb{R}$ be a continuous piecewise linear function with break points $\big\{\frac{k}{2J^d} : k = 0, 1, \cdots, 2J^d\big\}$ and the values of g at these break points satisfy the following properties:

- At the break point 1, let $g(1) = f(\frac{[J-1]^d}{J})$, where $[J-1]^d = [J-1, J-1, \cdots, J-1]^T \in \{0, 1, \cdots, J-1\}^d$;
- The values of g at the break points in $\mathcal{A}_1$ are set as

$$g(\tfrac{\theta_d}{2J^d} + \sum_{i=1}^{d-1} \tfrac{\theta_i}{J^i}) = \bar{f}(\tfrac{\boldsymbol{\theta}}{J}), \quad \text{for any } \boldsymbol{\theta} \in \{0, 1, \cdots, J-1\}^d; \tag{2.11}$$

- The values of g at the break points in $\mathcal{A}_2$ are assigned to reduce the variation of g. To this end, for each $\boldsymbol{\theta} \in \{0, 1, \cdots, J-1\}^d$, let g be linear on $[\xi_1^{\boldsymbol{\theta}}, \xi_3^{\boldsymbol{\theta}}]$, where $\xi_1^{\boldsymbol{\theta}} = \frac{J-1}{2J^d} + \sum_{i=1}^{d-1} \frac{\theta_i}{J^i} \in \mathcal{A}_1$ and $\xi_3^{\boldsymbol{\theta}} = \frac{1}{J^{d-1}} + \sum_{i=1}^{d-1} \frac{\theta_i}{J^i} \in \mathcal{A}_1 \cup \{1\}$, which requires that $g(\xi_2^{\boldsymbol{\theta}}) = \frac{(\xi_2^{\boldsymbol{\theta}} - \xi_1^{\boldsymbol{\theta}}) g(\xi_3^{\boldsymbol{\theta}}) + (\xi_3^{\boldsymbol{\theta}} - \xi_2^{\boldsymbol{\theta}}) g(\xi_1^{\boldsymbol{\theta}})}{\xi_3^{\boldsymbol{\theta}} - \xi_1^{\boldsymbol{\theta}}}$, where $\xi_2^{\boldsymbol{\theta}} = \frac{1}{2J^{d-1}} + \frac{\theta_d}{2J^d} + \sum_{i=1}^{d-1} \frac{\theta_i}{J^i} \in \mathcal{A}_2$.

Apparently, such a function g exists (see Figure 8 for an example) and satisfies

$$\big| g(\tfrac{k}{2J^d}) - g(\tfrac{k-1}{2J^d}) \big| \le \max\big\{ \omega_f(\tfrac{1}{J}), \omega_f(\sqrt{d})/J \big\} \le \omega_f(\tfrac{\sqrt{d}}{J}), \quad \text{for } k = 1, 2, \cdots, 2J^d,$$

and

$$0 \le g(\tfrac{k}{2J^d}) \le 2\omega_f(\sqrt{d}), \quad \text{for } k = 0, 1, \cdots, 2J^d.$$

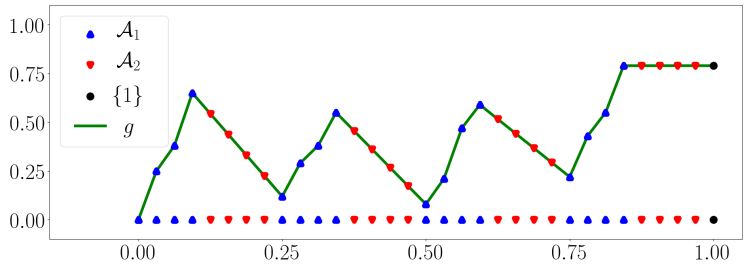

FIG. 8. *An illustration of* $\mathcal{A}_1$, $\mathcal{A}_2$, $\{1\}$, *and* g *for* $d=2$ *and* $J=4$.

Step 5: Construct a sub-FNN ϕ_3 mapping $\mathcal{A}_1$ to $\big\{\bar{f}(x): x\in\{0,1,\cdots,J-1\}^d/J\big\}$.

Since $2J^d = 2\big(\lfloor N^{1/d}\rfloor^2\lfloor L^{2/d}\rfloor\big)^d \le 2\big(N^2L^2\big) \le N^2\widetilde{L}^2$, where $\widetilde{L}=2L$, by Proposition 2.3, there exists $\psi\in \mathrm{NN}(\#\text{input}=1;\ \text{maxwidth}\le 12N+14;\ \#\text{layer}\le 4\widetilde{L}+10) = \mathrm{NN}(\#\text{input}=1;\ \text{maxwidth}\le 12N+14;\ \#\text{layer}\le 8L+10)$ such that

$$|\psi(k)-g(\tfrac{k}{2J^d})|\le \omega_f(\tfrac{\sqrt{d}}{J}),\quad \text{for } k=0,1,\cdots,2J^d-1,$$

and

$$0\le\psi(t)\le \max\{g(\tfrac{k}{2J^d}): k=0,1,\cdots,2J^d-1\}\le 2\omega_f(\sqrt{d}),\quad \text{for any } t\in\mathbb{R}.$$

Define $\phi_3(t):=\psi(2J^dt)$ for any $t\in\mathbb{R}$. It follows that

$$|\phi_3(\tfrac{k}{2J^d})-g(\tfrac{k}{2J^d})| = |\psi(k)-g(\tfrac{k}{2J^d})|\le \omega_f(\tfrac{\sqrt{d}}{J}),\quad \text{for } k=0,1,\cdots,2J^d-1, \tag{2.12}$$

and

$$0\le\phi_3(t)=\psi(2J^dt)\le 2\omega_f(\sqrt{d}),\quad \text{for any } t\in\mathbb{R}. \tag{2.13}$$

Step 6: Construct the final target FNN $\phi=\phi_3\circ\phi_2\circ\boldsymbol{\phi}_1+f(0)-\omega_f(\sqrt{d})$.

Define $\phi:=\bar{\phi}+f(0)-\omega_f(\sqrt{d})$, where $\bar{\phi}:=\phi_3\circ\phi_2\circ\boldsymbol{\phi}_1$. By Proposition 2.2, ϕ_1 constructed in Step 2 is in $\mathrm{NN}(\#\text{input}=1;\ \text{widthvec}=[2\lfloor N^{1/d}\rfloor, 4\lfloor N^{1/d}\rfloor\lfloor L^{2/d}\rfloor-1])\subseteq \mathrm{NN}(\#\text{input}=1;\ \text{maxwidth}\le 8\lfloor N^{1/d}\rfloor+4;\ \#\text{layer}\le \lfloor L^{2/d}\rfloor+2])$. Hence, for $\boldsymbol{\phi}_1$ constructed in Step 2 and ϕ_2 constructed in Step 3, we have $\phi_2\circ\boldsymbol{\phi}_1\in\mathrm{NN}(\#\text{input}=d;\ \text{maxwidth}\le 8d\lfloor N^{1/d}\rfloor+4d;\ \#\text{layer}\le\lfloor L^{2/d}\rfloor+2])$. It follows that $\phi=\phi_3\circ\phi_2\circ\boldsymbol{\phi}_1+f(0)-\omega_f(\sqrt{d})\in\mathrm{NN}(\#\text{input}=d;\ \text{maxwidth}\le\max\{8d\lfloor N^{1/d}\rfloor+4d,\ 12N+14\};\ \#\text{layer}\le 9L+12)$ as desired.

Now let us estimate the approximation error. By (2.11), (2.12), and the definition of ϕ_1, for any $\boldsymbol{x}=[x_1,x_2,\cdots,x_d]^T\in Q_{\boldsymbol{\theta}}$, we have

$$|\bar{f}(\boldsymbol{x})-\bar{\phi}(\boldsymbol{x})| = \big|\bar{f}(\boldsymbol{x})-\phi_3\big(\tfrac{1}{2J^d}\phi_1(x_d)+\sum_{i=1}^{d-1}\tfrac{1}{J^i}\phi_1(x_i)\big)\big| = \big|\bar{f}(\boldsymbol{x})-\phi_3\big(\tfrac{\theta_d}{2J^d}+\sum_{i=1}^{d-1}\tfrac{\theta_i}{J^i}\big)\big|,$$

which is bounded by

$$\big|\bar{f}(\boldsymbol{x})-g\big(\tfrac{\theta_d}{2J^d}+\sum_{i=1}^{d-1}\tfrac{\theta_i}{J^i}\big)\big|+\big|g\big(\tfrac{\theta_d}{2J^d}+\sum_{i=1}^{d-1}\tfrac{\theta_i}{J^i}\big)-\phi_3\big(\tfrac{\theta_d}{2J^d}+\sum_{i=1}^{d-1}\tfrac{\theta_i}{J^i}\big)\big|\le\big|\bar{f}(\boldsymbol{x})-\bar{f}(\tfrac{\boldsymbol{\theta}}{J})\big|+\omega_f(\tfrac{\sqrt{d}}{J})=2\omega_f(\tfrac{\sqrt{d}}{J}).$$

In sum,

$$|f(\boldsymbol{x})-\phi(\boldsymbol{x})| = |\bar{f}(\boldsymbol{x})-\bar{\phi}(\boldsymbol{x})|\le 2\omega_f(\tfrac{\sqrt{d}}{J})\le 2\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d}),\quad \text{for any } \boldsymbol{x}\in[0,1]^d\backslash\mathcal{H}, \tag{2.14}$$

where the last inequality comes from the fact $J = \lfloor N^{1/d}\rfloor^2\lfloor L^{2/d}\rfloor \geq \frac{N^{2/d}L^{2/d}}{8}$ for any $N, L \in \mathbb{N}^+$. By (2.9), $\mu(\mathcal{H}) \leq Jd\delta \leq \eta$. It remains to show the upper bound of ϕ. By (2.13) and the definition of $\bar{\phi}$, it holds that $0 \leq \bar{\phi}(\boldsymbol{x}) \leq 2\omega_f(\sqrt{d})$, for any $\boldsymbol{x} \in \mathbb{R}^d$. Together with $\phi = \bar{\phi} + f(0) - \omega_f(\sqrt{d})$, it follows that $\|\phi\|_{L^\infty([0,1]^d)} \leq |f(0)| + \omega_f(\sqrt{d})$. Thus, we finish the proof for the case $d \geq 2$.

2.5. Proofs of Propositions in Section 2.2. This section proves Propositions 2.1 to 2.3 in Section 2.2. In fact, Proposition 2.1 is a part of Lemma 2.2 in [57]. For the purpose of being self-contained, we present it as follows.

LEMMA 2.4 (Lemma 2.2 of [57]). *For any $m, n \in \mathbb{N}^+$, given any $m(n+1)+1$ samples $(x_i, y_i) \in \mathbb{R}^2$ with $x_0 < x_1 < x_2 < \cdots < x_{m(n+1)}$ and $y_i \geq 0$ for $i = 0, 1, \cdots, m(n+1)$, there exists $\phi \in \mathrm{NN}(\#\text{input} = 1; \text{widthvec} = [2m, 2n+1])$ satisfying the following conditions.*

1. *$\phi(x_i) = y_i$ for $i = 0, 1, \cdots, m(n+1)$;*
2. *ϕ is linear on each interval $[x_{i-1}, x_i]$ for $i \notin \{(n+1)j : j = 1, 2, \cdots, m\}$;*
3. *$sup_{x\in[x_0,\, x_{m(n+1)}]}|\phi(x)| \leq 3\max_{i\in\{0,1,\cdots,m(n+1)\}} y_i \prod_{k=1}^{n}\Big(1 + \frac{\max\{x_{j(n+1)+n} - x_{j(n+1)+k-1} : j = 0, 1, \cdots, m-1\}}{\min\{x_{j(n+1)+k} - x_{j(n+1)+k-1} : j = 0, 1, \cdots, m-1\}}\Big)$.*

The key idea to prove Proposition 2.2 is to re-assemble $\mathcal{O}(L)$ sub-FNNs with width $\mathcal{O}(N)$ and depth $\mathcal{O}(1)$ in the shallower FNN in the middle of Figure 6 to form a deeper one with width $\mathcal{O}(N)$ and depth $\mathcal{O}(L)$ on the right of Figure 6.

Proof of Proposition 2.2. For any $\phi \in \mathrm{NN}(\#\text{input} = 1; \text{widthvec} = [N, NL])$, the architecture of ϕ can be described as

$$x \xrightarrow{\boldsymbol{W}_1,\ \boldsymbol{b}_1} \boldsymbol{g} \xrightarrow{\boldsymbol{W}_2,\ \boldsymbol{b}_2} \boldsymbol{h} \xrightarrow{\boldsymbol{W}_3,\ \boldsymbol{b}_3} \phi(x),$$

where $\boldsymbol{g}$ and $\boldsymbol{h}$ are the output of the first hidden layer and the second hidden layer, respectively. Note that

$$\boldsymbol{g} = \sigma(\boldsymbol{W}_1 x + \boldsymbol{b}_1), \quad \boldsymbol{h} = \sigma(\boldsymbol{W}_2\boldsymbol{g} + \boldsymbol{b_2}), \quad \text{and} \quad \phi(x) = \boldsymbol{W}_3\boldsymbol{h} + \boldsymbol{b_3}. \tag{2.15}$$

We can evenly divide $\boldsymbol{h}$, $\boldsymbol{b}_2$, $\boldsymbol{W}_2$, and $\boldsymbol{W}_3$ into L parts as follows:

$$\begin{bmatrix} \boldsymbol{h}_1 \\ \boldsymbol{h}_2 \\ \vdots \\ \boldsymbol{h}_L \end{bmatrix}, \quad \begin{bmatrix} \boldsymbol{b}_{2,1} \\ \boldsymbol{b}_{2,2} \\ \vdots \\ \boldsymbol{b}_{2,L} \end{bmatrix}, \quad \begin{bmatrix} \boldsymbol{W}_{2,1} \\ \boldsymbol{W}_{2,2} \\ \vdots \\ \boldsymbol{W}_{2,L} \end{bmatrix},$$

and $[\boldsymbol{W}_{3,1}, \boldsymbol{W}_{3,2}, \cdots, \boldsymbol{W}_{3,L}]$, where $\boldsymbol{h}_\ell,\ \boldsymbol{b}_{2,\ell} \in \mathbb{R}^N$, $\boldsymbol{W}_{2,\ell} \in \mathbb{R}^{N\times N}$, and $\boldsymbol{W}_{3,\ell} \in \mathbb{R}^{1\times N}$ for $\ell = 1, 2, \cdots, L$. Define $y_\ell := \boldsymbol{W}_{3,\ell}\boldsymbol{h}_\ell$ and $s_\ell := \sum_{j=1}^{\ell} y_j$ for $\ell = 1, 2, \cdots, L$. Note that $\boldsymbol{g} \geq 0$ since it is the output of the ReLU function. Hence, it is easy to check that the desired deep FNN can be constructed as follows:

$$x \to \boldsymbol{g} \to \begin{bmatrix} \boldsymbol{h}_1 \\ \boldsymbol{g} \end{bmatrix} \to \begin{bmatrix} \sigma(y_1) \\ \sigma(-y_1) \\ \boldsymbol{h}_2 \\ \boldsymbol{g} \end{bmatrix} \to \begin{bmatrix} \sigma(s_1) \\ \sigma(-s_1) \\ \sigma(y_2) \\ \sigma(-y_2) \\ \boldsymbol{h}_3 \\ \boldsymbol{g} \end{bmatrix} \to \cdots \to \begin{bmatrix} \sigma(s_{L-2}) \\ \sigma(-s_{L-2}) \\ \sigma(y_{L-1}) \\ \sigma(-y_{L-1}) \\ \boldsymbol{h}_L \\ \boldsymbol{g} \end{bmatrix} \to \begin{bmatrix} \sigma(s_{L-1}) \\ \sigma(-s_{L-1}) \\ \sigma(y_L) \\ \sigma(-y_L) \end{bmatrix} \to \phi(x),$$

where "$\to$" represents the composition of a ReLU activation function and an appropriate linear transform with weights and bias from the transforms in (2.15) up to the change of their signs. It is clear that the FNN with above architecture has width $2N + 4$ and depth $L + 2$. So, we finish the proof. ∎

The proof of Proposition 2.3 is based on the bit extraction technique in [5, 25]. In fact, we modify this technique to extract the sum of many bits rather than one bit and this modification can be summarized in Lemma 2.5 and 2.6 below.

LEMMA 2.5. *For any $L \in \mathbb{N}^+$, there exists $\phi \in \mathrm{NN}(\#\text{input} = 2;\ \text{maxwidth} \le 7;\ \#\text{layer} \le 2L+1)$ such that for any $x = \mathrm{bin}\,0.x_1x_2\cdots x_L$, we have $\phi(x,\ell) = \sum_{j=1}^{\ell} x_j$ for $\ell = 1, 2, \cdots, L$.*

Proof of Lemma 2.5. For any $x = \mathrm{bin}\,0.x_1x_2\cdots x_L$, we define $\xi_j \coloneqq \mathrm{bin}\,0.x_jx_{j+1}\cdots x_L$ for $j = 1, 2, \cdots, L$, and

$$\mathcal{T}(t) \coloneqq \begin{cases} 1, & t \ge 0, \\ 0, & t < 0. \end{cases}$$

Then $x_j = \mathcal{T}(\xi_j - 1/2)$ for $j = 1, 2, \cdots, L$, and $\xi_{j+1} = 2\xi_j - x_j$ for $j = 1, 2, \cdots, L-1$.

Next, ReLU FNNs will be constructed to represent $\mathcal{T}(\cdot)$ and identity maps in the above discussion. In fact, $\mathcal{T}(t) = \sigma(t/\delta + 1) - \sigma(t/\delta)$ for any $t \notin (-\delta, 0)$. Hence, if we set $\delta = 1/2 - \sum_{j=2}^{L} 2^{-j} = 2^{-L}$, then

$$x_j = \mathcal{T}(\xi_j - 1/2) = \sigma\big((\xi_j - 1/2)/\delta + 1\big) - \sigma\big((\xi_j - 1/2)/\delta\big) \tag{2.16}$$

for $j = 1, 2, \cdots, L$. Note that $2\xi_j - x_j$ is positive for $j = 1, 2, \cdots, L-1$. Hence,

$$\xi_{j+1} = 2\xi_j - x_j = \sigma(2\xi_j - x_j), \quad \text{for } j = 1, 2, \cdots, L-1. \tag{2.17}$$

Now let us establish a formula to represent $\sum_{j=1}^{\ell} x_j$ in a form of a ReLU FNN as follows:

$$\sum_{j=1}^{\ell} x_j = \sum_{j=1}^{L} \mathcal{T}(\ell - j)x_j = \sum_{j=1}^{L} \big(\sigma(\ell - j + 1) - \sigma(\ell - j)\big)x_j, \quad \text{for } \ell = 1, 2, \cdots, L,$$

where the last equality comes from the fact $\mathcal{T}(n) = \sigma(n+1) - \sigma(n)$ for any integer n. The fact that $t_1t_2 = \sigma(t_1 + t_2 - 1)$ for any $t_1, t_2 \in \{0, 1\}$ implies

$$\sum_{j=1}^{\ell} x_j = \sum_{j=1}^{L} \big(\sigma(\ell - j + 1) - \sigma(\ell - j)\big)x_j = \sum_{j=1}^{L} \sigma\big(\sigma(\ell - j + 1) - \sigma(\ell - j) + x_j - 1\big). \tag{2.18}$$

Define

$$a_j \coloneqq \sigma\big(\sigma(\ell - j + 1) - \sigma(\ell - j) + x_j - 1\big), \quad \text{and} \quad s_j \coloneqq \sum_{i=1}^{j} a_i, \quad \text{for } j = 1, 2, \cdots, L. \tag{2.19}$$

By the definitions of a_j, s_j, x_j, and ξ_j in Equation (2.16), (2.17), (2.18), and (2.19), it is easy to construct a ReLU FNN in $\mathrm{NN}(\#\text{input} = 2;\ \text{maxwidth} \le 7;\ \#\text{layer} \le 2L+1)$ outputting $\sum_{j=1}^{\ell} x_j = \sum_{j=1}^{L} a_j = s_L$ given the input (x, ℓ) with $x = \mathrm{bin}\,0.x_1x_2\cdots x_L$ and $\ell \in \{1, 2, \cdots, L\}$. The skeleton of the desired FNN is shown in Figure 9. Hence, we finish the proof. ∎

Next, we introduce Lemma 2.6 as an advanced version of Lemma 2.5.

LEMMA 2.6. *For any $N, L \in \mathbb{N}^+$, any $\theta_{m,\ell} \in \{0, 1\}$ for $m = 0, 1, \cdots, M-1$, $\ell = 0, 1, \cdots, L-1$, where $M = N^2L$, there exists $\phi \in \mathrm{NN}(\#\text{input} = 2;\ \text{maxwidth} \le 4N+5;\ \#\text{layer} \le 3L+4)$ such that $\phi(m, \ell) = \sum_{j=0}^{\ell} \theta_{m,j}$, for $m = 0, 1, \cdots, M-1$, $\ell = 0, 1, \cdots, L-1$.*

Proof of Lemma 2.6. Define $y_m \coloneqq \mathrm{bin}\,0.\theta_{m,0}\theta_{m,1}\cdots\theta_{m,L-1}$ for $m = 0, 1, \cdots, M-1$. By Proposition 2.1, for the sample set $\{(m, y_m) : m = 0, 1, \cdots, M\}$, whose cardinality is $M + 1 = N\big((NL-1)+1\big) + 1$, there exists $\phi_1 \in \mathrm{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 2(NL-1)+1]) = \mathrm{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 2NL-1])$ such that $\phi_1(m) = y_m$, for $m = 0, 1, \cdots, M-1$. By Lemma 2.5, there exists $\phi_2 \in \mathrm{NN}(\#\text{input} = 2;\ \text{maxwidth} \le 7;\ \#\text{layer} \le 2L+1)$ such that $\phi_2(x, \ell) = \sum_{j=1}^{\ell} x_j$ for $\ell = 1, 2, \cdots, L$, if $x = \mathrm{bin}\,0.x_1x_2\cdots x_L$.

Note that $\phi_2(\phi_1(m), \ell + 1) = \phi_2(y_m, \ell + 1) = \sum_{j=0}^{\ell} \theta_{m,j}$, for $m = 0, 1, \cdots, M-1$, $\ell = 0, 1, \cdots, L-1$. Define

$$\phi(t_1, t_2) \coloneqq \phi_2\Big(\sigma\big(\phi_1(t_1)\big), \sigma(t_2 + 1)\Big), \quad \text{for any } t_1, t_2 \in \mathbb{R}.$$

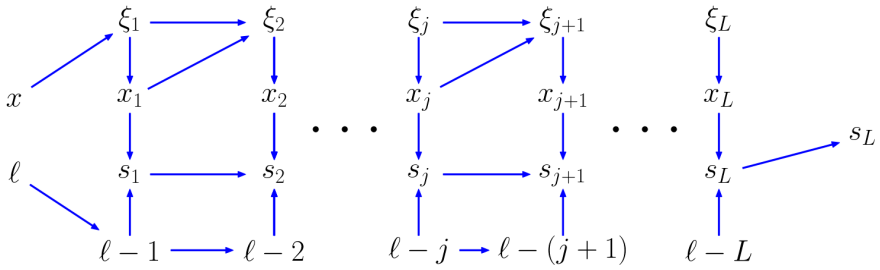

FIG. 9. *A skeleton of the target ReLU FNN to illustrate how to output* $\sum_{j=1}^{\ell} x_j = \sum_{j=1}^{L} a_j = s_L$ *given the input* (x,ℓ) *with* $x = \text{bin}\,0.x_1x_2\cdots x_L$ *and* $\ell \in \{1,2,\cdots,L\}$*, following the iterative definitions of* a_j*,* s_j*,* x_j*, and* ξ_j *in Equation* (2.16)*,* (2.17)*,* (2.18)*, and* (2.19)*.*

Then $\phi(m,\ell) = \phi_2\Big(\sigma\big(\phi_1(m)\big),\sigma(\ell+1)\Big) = \phi_2(\phi_1(m),\ell+1) = \phi_2(y_m,\ell+1) = \sum_{j=0}^{\ell}\theta_{m,j}$, for $m = 0,1,\cdots,M-1$, $\ell = 0,1,\cdots,L-1$. And the architecture of ϕ is

$$\begin{bmatrix} t_1 \\ t_2 \end{bmatrix} \to \begin{bmatrix} \sigma\big(\phi_1(t_1)\big) \\ \sigma(t_2+1) \end{bmatrix} \to \phi_2\Big(\sigma\big(\phi_1(t_1)\big),\sigma(t_2+1)\Big) = \phi(t_1,t_2).$$

By Proposition 2.2, $\phi_1 \in \text{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 2NL-1]) \subseteq \text{NN}(\#\text{input} = 1;\ \text{maxwidth} \le 4N+4;\ \#\text{layer} \le L+2)$ implies $\phi \in \text{NN}(\#\text{input} = 2;\ \text{maxwidth} \le 4N+5;\ \#\text{layer} \le 3L+4)$ as we desire. Therefore, we have finished the proof. ∎

Next, we apply Lemma 2.6 to prove Lemma 2.7 below, which is a key intermediate conclusion to prove Proposition 2.3.

LEMMA 2.7. *For any* $\varepsilon > 0$*,* $L, N \in \mathbb{N}^+$*, any sample set* $\{y_{m,\ell} \ge 0 : m = 0,1,\cdots,M-1,\ \ell = 0,1,\cdots,L-1\}$ *with* $|y_{m,\ell} - y_{m,\ell-1}| \le \varepsilon$*, for* $m = 0,1,\cdots,M-1,\ \ell = 1,2,\cdots,L-1$*, where* $M = N^2L$*, there exists* $\phi \in \text{NN}(\#\text{input} = 2;\ \text{maxwidth} \le 12N+14;\ \#\text{layer} \le 3L+7)$ *such that*

1. $|\phi(m,\ell) - y_{m,\ell}| \le \varepsilon$*, for* $m = 0,1,\cdots,M-1,\ \ell = 0,1,\cdots,L-1$*;*
2. $0 \le \phi(t_1,t_2) \le \max\{y_{m,\ell} : m = 0,1,\cdots,M-1,\ \ell = 0,1,\cdots,L-1\}$*, for any* $t_1, t_2 \in \mathbb{R}$*.*

Proof of Lemma 2.7. Define

$$a_{m,\ell} := \lfloor y_{m,\ell}/\varepsilon \rfloor, \quad \text{for } m = 0,1,\cdots,M-1,\ \ell = 0,1,\cdots,L-1.$$

We will construct a sub-FNN mapping the index (m,ℓ) to $a_{m,\ell}\varepsilon$ for $m = 0,1,\cdots,M-1$, $\ell = 0,1,\cdots,L-1$. Set $b_{m,0} := 0$ and $b_{m,\ell} := a_{m,\ell} - a_{m,\ell-1}$ for $m = 0,1,\cdots,M-1$, $\ell = 1,\cdots,L-1$. Since $|y_{m,\ell} - y_{m,\ell-1}| \le \varepsilon$ for all m and ℓ, we have $b_{m,\ell} \in \{-1,0,1\}$. Hence, there exist $c_{m,\ell}$ and $d_{m,\ell} \in \{0,1\}$ such that $b_{m,\ell} = c_{m,\ell} - d_{m,\ell}$. It follows that

$$a_{m,\ell} = a_{m,0} + \sum_{j=1}^{\ell}(a_{m,j} - a_{m,j-1}) = a_{m,0} + \sum_{j=1}^{\ell} b_{m,j} = a_{m,0} + \sum_{j=0}^{\ell} b_{m,j} = a_{m,0} + \sum_{j=0}^{\ell} c_{m,j} - \sum_{j=0}^{\ell} d_{m,j}.$$

For the sample set $\{(m, a_{m,0}) : m = 0,1,\cdots,M-1\} \cup \{(M,0)\}$, whose cardinality is $M+1 = N\cdot\big((NL-1)+1\big)+1$, by Proposition 2.1, there exists $\phi_1 \in \text{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 2(NL-1)+1]) = \text{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 2NL-1])$ such that $\phi_1(m) = a_{m,0}$ for $m = 0,1,\cdots,M-1$. By Lemma 2.6, there exist $\phi_2, \phi_3 \in \text{NN}(\#\text{input} = 2;\ \text{maxwidth} \le 4N+5;\ \#\text{layer} \le 3L+4)$ such that $\phi_2(m,\ell) = \sum_{j=0}^{\ell} c_{m,j}$ and $\phi_3(m,\ell) = \sum_{j=0}^{\ell} d_{m,j}$ for

$m = 0, 1, \cdots, M-1,\ \ell = 0, 1, \cdots, L-1$. Hence, it holds that

$$a_{m,\ell} = a_{m,0} + \sum_{j=0}^{\ell} c_{m,j} - \sum_{j=0}^{\ell} d_{m,j} = \phi_1(m) + \phi_2(m,\ell) - \phi_3(m,\ell),$$

for $m = 0, 1, \cdots, M-1,\ \ell = 0, 1, \cdots, L-1$. So, define

$$\bar{\phi}(t_1, t_2) \coloneqq \Big(\sigma\big(\phi_1(t_1)\big) + \sigma\big(\phi_2(t_1,t_2)\big) - \sigma\big(\phi_3(t_1,t_2)\big)\Big)\varepsilon, \quad \text{for any } t_1, t_2 \in \mathbb{R}.$$

It follows that, for $m = 0, 1, \cdots, M-1,\ \ell = 0, 1, \cdots, L-1$, we have

$$\begin{aligned}
\bar{\phi}(m,\ell) &= \Big(\sigma\big(\phi_1(m)\big) + \sigma\big(\phi_2(m,\ell)\big) - \sigma\big(\phi_3(m,\ell)\big)\Big)\varepsilon \\
&= \big(\phi_1(m) + \phi_2(m,\ell) - \phi_3(m,\ell)\big)\varepsilon \\
&= \big(a_{m,0} + \sum_{j=0}^{\ell} c_{m,j} - \sum_{j=0}^{\ell} d_{m,j}\big)\varepsilon \\
&= a_{m,\ell}\varepsilon.
\end{aligned}$$

And the architecture of $\bar{\phi}$ is

$$\begin{bmatrix} t_1 \\ t_2 \end{bmatrix} \to \begin{bmatrix} \sigma(\phi_1(t_1)) \\ \sigma(\phi_2(t_1,t_2)) \\ \sigma(\phi_3(t_1,t_2)) \end{bmatrix} \to \Big(\sigma\big(\phi_1(t_1)\big) + \sigma\big(\phi_2(t_1,t_2)\big) - \sigma\big(\phi_3(t_1,t_2)\big)\Big)\varepsilon = \bar{\phi}(t_1,t_2).$$

By Proposition 2.2, $\phi_1 \in \mathrm{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 2NL-1]) \subseteq \mathrm{NN}(\#\text{input} = 1;\ \text{maxwidth} \le 4N+4;\ \#\text{layer} \le L+2)$, which means $\bar{\phi} \in \mathrm{NN}(\#\text{input} = 2;\ \text{maxwidth} \le 12N+14;\ \#\text{layer} \le 3L+5)$. Since $\bar{\phi}$ may not be bounded appropriately, we define

$$\phi(t_1, t_2) \coloneqq \min\{\sigma(\bar{\phi}(t_1,t_2)),\, y_{\max}\}, \quad \text{for any } t_1, t_2 \in \mathbb{R},$$

where

$$y_{\max} \coloneqq \max\{y_{m,\ell} : m = 0, 1, \cdots, M-1,\ \ell = 0, 1, \cdots, L-1\}.$$

It is clear that

$$0 \le \phi(t_1,t_2) \le y_{\max} = \max\{y_{m,\ell} : m = 0, 1, \cdots, M-1,\ \ell = 0, 1, \cdots, L-1\}, \quad \text{for any } t_1, t_2 \in \mathbb{R},$$

and that

$$\phi(m,\ell) = \min\{\sigma(\bar{\phi}(m,\ell)),\, y_{\max}\} = \min\{a_{m,\ell}\varepsilon,\, y_{\max}\} = a_{m,\ell}\varepsilon,$$

for $m = 0, 1, \cdots, M-1,\ \ell = 0, 1, \cdots, L-1$.

It follows that

$$|\phi(m,\ell) - y_{m,\ell}| = |a_{m,\ell}\varepsilon - y_{m,\ell}| = \big|\lfloor y_{m,\ell}/\varepsilon \rfloor \varepsilon - y_{m,\ell}\big| \le \varepsilon,$$

for $m = 0, 1, \cdots, M-1,\ \ell = 0, 1, \cdots, L-1$. It remains to establish the architecture of ϕ. Since $\min\{t_1, t_2\} = \frac{t_1+t_2-|t_1-t_2|}{2} = \frac{\sigma(t_1+t_2)-\sigma(-t_1-t_2)-\sigma(t_1-t_2)-\sigma(t_2-t_1)}{2}$ for any $t_1, t_2 \in \mathbb{R}$, the architecture of ϕ is

$$\begin{bmatrix} t_1 \\ t_2 \end{bmatrix} \to \sigma(\bar{\phi}(t_1,t_2)) \to \begin{bmatrix} \sigma\big(\sigma(\bar{\phi}(t_1,t_2)) + y_{\max}\big) \\ \sigma\big(-\sigma(\bar{\phi}(t_1,t_2)) - y_{\max}\big) \\ \sigma\big(\sigma(\bar{\phi}(t_1,t_2)) - y_{\max}\big) \\ \sigma\big(-\sigma(\bar{\phi}(t_1,t_2)) + y_{\max}\big) \end{bmatrix} \to \min\{\sigma(\bar{\phi}(t_1,t_2)),\, y_{\max}\} = \phi(t_1,t_2).$$

Therefore, $\phi \in \mathrm{NN}(\#\text{input} = 2;\ \text{maxwidth} \le 12N + 14;\ \#\text{layer} \le 3L + 7)$. Hence, we finish the proof. ∎

Finally, we apply Lemma 2.7 to prove Proposition 2.3.

Proof of Proposition 2.3. Let $M = N^2L$, then we may assume $K = ML$ since we can assume $y_{K-1} = y_K = y_{K+1} = \cdots = y_{ML-1}$ if $K < ML$. For the sample set $\{(mL, m) : m = 0, 1, \cdots, M\} \cup \{(mL + L - 1, m) : m = 0, 1, \cdots, M - 1\}$, whose cardinality is $2M + 1 = N \cdot \big((2NL - 1) + 1\big) + 1$, by Proposition 2.1 there exist $\phi_1 \in \mathrm{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 2(2NL - 1) + 1]) = \mathrm{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 4NL - 1])$ such that

- $\phi_1(ML) = M$ and $\phi_1(mL) = \phi_1(mL + L - 1) = m$ for $m = 0, 1, \cdots, M - 1$;
- ϕ_1 is linear on each interval $[mL, mL + L - 1]$ for $m = 0, 1, \cdots, M - 1$.

Note that any number k in $\{0, 1, \ldots, K-1\}$ can be indexed as $k = mL + \ell$ for $m = 0, 1, \cdots, M-1$, $\ell = 0, 1, \cdots, L-1$. Hence, we can denote $y_k = y_{mL+\ell}$ as $y_{m,\ell}$ for $m = 0, 1, \cdots, M - 1$, $\ell = 0, 1, \cdots, L - 1$, which gives

$$\begin{cases} \phi_1(k) = m, \\ k - L\phi_1(k) = \ell. \end{cases} \tag{2.20}$$

Then by Lemma 2.7, there exists $\phi_2 \in \mathrm{NN}(\#\text{input} = 2;\ \text{maxwidth} \le 12N + 14;\ \#\text{layer} \le 3L + 7)$ such that

$$|\phi_2(m, \ell) - y_{m,\ell}| \le \varepsilon, \quad \text{for } m = 0, 1, \cdots, M - 1,\ \ell = 0, 1, \cdots, L - 1, \tag{2.21}$$

and

$$0 \le \phi_2(t_1, t_2) \le \max\{y_{m,\ell} : m = 0, 1, \cdots, M - 1,\ \ell = 0, 1, \cdots, L - 1\}, \quad \text{for any } t_1, t_2 \in \mathbb{R}. \tag{2.22}$$

Define $\phi(t) \coloneqq \phi_2\Big(\sigma\big(\phi_1(t)\big), \sigma(t) - L\sigma\big(\phi_1(t)\big)\Big)$ for any $t \in \mathbb{R}$. By Equation (2.20) and (2.21), it holds that $|\phi(k) - y_k| = \Big|\phi_2\Big(\sigma\big(\phi_1(k)\big), \sigma(k) - L\sigma\big(\phi_1(k)\big)\Big) - y_k\Big| = |\phi_2(m, \ell) - y_{m,\ell}| \le \varepsilon$ for $k \in \{0, 1, \cdots, K - 1\}$. By Equation (2.22), for any $t \in \mathbb{R}$,

$$\begin{aligned} 0 \le \phi(t) &= \phi_2\Big(\sigma\big(\phi_1(t)\big), \sigma(t) - L\sigma\big(\phi_1(t)\big)\Big) \\ &\le \max\{y_{m,\ell} : m = 0, 1, \cdots, M - 1,\ \ell = 0, 1, \cdots, L - 1\} \\ &= \max\{y_k : k = 0, 1, \cdots, K - 1\}. \end{aligned}$$

And the architecture of ϕ is

$$t \to \begin{bmatrix} \sigma\big(\phi_1(t)\big) \\ \sigma(t) \end{bmatrix} \to \begin{bmatrix} \sigma\big(\phi_1(t)\big) \\ \sigma\Big(\sigma(t) - L\sigma\big(\phi_1(t)\big)\Big) \\ \sigma\Big(L\sigma\big(\phi_1(t)\big) - \sigma(t)\Big) \end{bmatrix} \to \phi_2\Big(\sigma\big(\phi_1(t)\big), \sigma(t) - L\sigma\big(\phi_1(t)\big)\Big) = \phi(t).$$

By Proposition 2.2, $\phi_1 \in \mathrm{NN}(\#\text{input} = 1;\ \text{widthvec} = [2N, 4NL - 1]) \subseteq \mathrm{NN}(\#\text{input} = 1;\ \text{maxwidth} \le 8N + 4;\ \#\text{layer} \le L + 2)$, which means $\phi \in \mathrm{NN}(\#\text{input} = 1;\ \text{maxwidth} \le 12N + 14;\ \#\text{layer} \le 4L + 10)$ as expected. So we finish the proof. ∎

We would like to remark that the key idea in the proof of Proposition 2.3 is the bit extraction technique in Lemma 2.5, which allows us to store L bits in a binary number $\mathrm{bin}\, 0.x_1x_2\cdots x_L$ and extract each bit x_i. The extraction operator can be efficiently carried out via a deep ReLU neural network demonstrating the power of depth.

3. Optimality of the Constructive Approximation. This section will show that the approximation rate in Theorem 1.1 is tight and there is no room to improve for the function class $\mathrm{Lip}(\nu, \alpha, d)$. First, we show that the approximation rate in Theorem 1.1 for the function class $\mathrm{Lip}(\nu, \alpha, d)$ cannot be improved to $\mathcal{O}(\omega_f(N^{-(2/d+\rho)}L^{-(2/d+\rho)}))$ in the $L^\infty([0,1]^d)$-norm for any $\rho > 0$. This means that the approximation rate in Theorem 1.1 is "nearly" tight, though the norm in Theorem 1.1 is different to the $L^\infty([0,1]^d)$-norm. Second, to finally prove the tightness using the same norm, we introduce a special norm denoted as $L^{p,\infty}([0,1]^d)$-norm for $p \in [1, \infty)$ later. We will show that Theorem 1.1 is valid in the $L^{p,\infty}([0,1]^d)$-norm and the approximation rate cannot be improved to $\mathcal{O}(\omega_f(N^{-(2/d+\rho)}L^{-(2/d+\rho)}))$ in the $L^{p,\infty}([0,1]^d)$-norm as well.

3.1. Unachievable Approximation Rate $\mathcal{O}(\omega_f(N^{-(2/d+\rho)}L^{-(2/d+\rho)}))$. Now we show that the approximation rate in Theorem 1.1 is "nearly" tight for the function class $\mathrm{Lip}(\nu,\alpha,d)$ by presenting an asymptotic unachievable approximation rate $\mathcal{O}(\omega_f(N^{-(2/d+\rho)}L^{-(2/d+\rho)}))$ for any $\rho>0$ in Theorem 3.1 below.

THEOREM 3.1. *Given any $\rho>0$ and $C>0$, there exists $f\in\mathrm{Lip}(\nu,\alpha,d)$ such that, for any $J_0>0$, there exist $N,L\in\mathbb{N}$ with $NL\geq J_0$ satisfying*

$$\inf_{\phi\in\mathrm{NN}(\#\mathrm{input}=d;\,\mathrm{maxwidth}\leq N;\,\#\mathrm{layer}\leq L)}\|\phi-f\|_{L^\infty([0,1]^d)}\geq C\nu N^{-(2\alpha/d+\rho)}L^{-(2\alpha/d+\rho)}.$$

In fact, we can show a stronger result than Theorem 3.1. Under the same conditions as in Theorem 3.1, for any $\mathcal{H}\in[0,1]^d$ with $\mu(\mathcal{H})\leq 2^{-(d+K^d+1)}K^{-d}$, where $K=\lfloor(NL)^{2/d+\rho/(2\alpha)}\rfloor$, it can be proved that

$$\inf_{\phi\in\mathrm{NN}(\#\mathrm{input}=d;\,\mathrm{maxwidth}\leq N;\,\#\mathrm{layer}\leq L)}\|\phi-f\|_{L^\infty([0,1]^d\setminus\mathcal{H})}\geq C\nu N^{-(2\alpha/d+\rho)}L^{-(2\alpha/d+\rho)}. \tag{3.1}$$

We will prove (3.1) by contradiction, then Theorem 3.1 holds as a consequence. The result of (3.1) will be used later in Section 3.2. Assuming Equation (3.1) is false, we have the following claim.

CLAIM 3.2. *There exist $\rho>0$ and $C>0$ such that given any $f\in\mathrm{Lip}(\nu,\alpha,d)$, there exists $J_0=J_0(\rho,C,f)>0$ such that, for any $N,L\in\mathbb{N}$ with $NL\geq J_0$, there exist $\phi\in\mathrm{NN}(\#\mathrm{input}=d;\,\mathrm{maxwidth}\leq N;\,\#\mathrm{layer}\leq L)$ and $\mathcal{H}\in[0,1]^d$ with $\mu(\mathcal{H})\leq 2^{-(d+K^d+1)}K^{-d}$, where $K=\lfloor(NL)^{2/d+\rho/(2\alpha)}\rfloor$, satisfying*

$$\|f-\phi\|_{L^\infty([0,1]^d\setminus\mathcal{H})}\leq C\nu N^{-(2\alpha/d+\rho)}L^{-(2\alpha/d+\rho)}.$$

Now let us disprove this claim to show Theorem 3.1 and Equation (3.1) are true.

Disproof of Claim 3.2. Without the loss of generality, we assume $\nu=1$; in the case of $\nu\neq 1$, the proof is similar. We will disprove Claim 3.2 using the VC dimension. Recall that the VC dimension of a class of functions is defined as the cardinality of the largest set of points that this class of functions can shatter. Denote the VC dimension of a function set $\mathcal{F}$ by $\mathrm{VCDim}(\mathcal{F})$. By [25], there exists $C_1>0$ such that

$$\mathrm{VCDim}\big(\mathrm{NN}(\#\mathrm{input}=d;\,\#\mathrm{parameter}\leq W;\,\#\mathrm{layer}\leq L)\big)\leq C_1WL\ln W. \tag{3.2}$$

Hence,

$$\begin{aligned}
&\mathrm{VCDim}\big(\mathrm{NN}(\#\mathrm{input}=d;\,\mathrm{maxwidth}\leq N;\,\#\mathrm{layer}\leq L)\big)\\
&\leq\mathrm{VCDim}\big(\mathrm{NN}(\#\mathrm{input}=d;\,\#\mathrm{parameter}\leq(LN+d+2)(N+1);\,\#\mathrm{layer}\leq L)\big)\\
&\leq C_1(LN+d+2)(N+1)L\ln\big((LN+d+2)(N+1)\big)\\
&\coloneqq b_u(N,L).
\end{aligned} \tag{3.3}$$

Then we will use Claim 3.2 to estimate a lower bound of

$$\mathrm{VCDim}\big(\mathrm{NN}(\#\mathrm{input}=d;\,\mathrm{maxwidth}\leq N;\,\#\mathrm{layer}\leq L)\big), \tag{3.4}$$

and this lower bound is asymptotically larger than $b_u(N,L)$, which leads to a contradiction.

More precisely, we will construct $\{f_\beta:\beta\in\mathscr{B}\}\subseteq\mathrm{Lip}(1,\alpha,d)$, which can shatter $b_\ell(N,L)\coloneqq K^d$ points, where $\mathscr{B}$ is a set defined later. Then by Claim 3.2, there exists $\{\phi_\beta:\beta\in\mathscr{B}\}$ such that this set can shatter $b_\ell(N,L)$ points. Finally, $b_\ell(N,L)=K^d=\lfloor(NL)^{2/d+\rho/(2\alpha)}\rfloor^d$ is asymptotically larger than $b_u(N,L)=C_1(LN+d+2)(N+1)L\ln\big((LN+d+2)(N+1)\big)$, which leads to a contradiction. More details can be found below.

Step 1: Construct $\{f_\beta:\beta\in\mathscr{B}\}\subseteq\mathrm{Lip}(1,\alpha,d)$ that scatters $b_\ell(N,L)$ points.

Let $Q(\boldsymbol{x}_0,r)\subseteq[0,1]^d$ be a cube, whose center and sidelength are $\boldsymbol{x}_0$ and r, respectively. Then we define a function ζ_Q on $[0,1]^d$ corresponding to $Q=Q(\boldsymbol{x}_0,r)\subseteq[0,1]^d$ such that:

- $\zeta_Q(\boldsymbol{x}_0) = (r/2)^\alpha/2$;
- $\zeta_Q(\boldsymbol{x}) = 0$ for any $\boldsymbol{x} \notin Q\backslash\partial Q$, where ∂Q is the boundary of Q;
- ζ_Q is linear on the line that connects $\boldsymbol{x}_0$ and $\boldsymbol{x}$ for any $\boldsymbol{x} \in \partial Q$.

Divide $[0,1]^d$ into K^d non-overlapping sub-cubes $\{Q_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta}}$ as follows:

$$Q_{\boldsymbol{\theta}} \coloneqq \Big\{\boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in [0,1]^d : x_i \in [\tfrac{\theta_i - 1}{K}, \tfrac{\theta_i}{K}],\ i = 1, 2, \cdots, d\Big\},$$

for any index vector $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_d]^T \in \{1, 2, \cdots, K\}^d$. Define

$$\mathscr{B} \coloneqq \big\{\beta : \beta \text{ is a map from } \{1, 2, \cdots, K\}^d \text{ to } \{-1, 1\}\big\}.$$

For each $\beta \in \mathscr{B}$, we define

$$f_\beta(\boldsymbol{x}) \coloneqq \sum_{\boldsymbol{\theta} \in \{1,2,\cdots,K\}^d} \beta(\boldsymbol{\theta}) \zeta_{Q_{\boldsymbol{\theta}}}(\boldsymbol{x}),$$

where $\zeta_{Q_{\boldsymbol{\theta}}}(\boldsymbol{x})$ is the associated function introduced just above. It is easy to check that $f_\beta \in \mathrm{Lip}(1, \alpha, d)$ and $\{f_\beta : \beta \in \mathscr{B}\}$ can shatter $b_\ell(N, L) = K^d$ points.

Step 2: Construct $\{\phi_\beta : \beta \in \mathscr{B}\}$ that scatters $b_\ell(N, L)$ points.

By Claim 3.2, there exist $\rho > 0$ and $C_2 > 0$ such that for any $f_\beta \in \{f_\beta : \beta \in \mathscr{B}\}$ there exists $J_\beta > 0$ such that for all $N, L \in \mathbb{N}$ with $NL \geq J_\beta$, there exist $\phi_\beta \in \mathrm{NN}(\#\text{input} = 1; \text{maxwidth} \leq N; \#\text{layer} \leq L)$ and $\mathcal{H}_\beta$ with $\mu(\mathcal{H}_\beta) \leq 2^{-(d+K^d+1)} K^{-d}$ such that

$$|f_\beta(\boldsymbol{x}) - \phi_\beta(\boldsymbol{x})| \leq C_2 (NL)^{-\alpha(2/d+\rho/\alpha)}, \quad \text{for any } \boldsymbol{x} \in [0,1]^d \backslash \mathcal{H}_\beta.$$

Set $\mathcal{H} = \cup_{\beta \in \mathscr{B}} \mathcal{H}_\beta$ and $J_1 = \max_{\beta \in \mathscr{B}} J_\beta$. Then it holds that

$$\mu(\mathcal{H}) \leq 2^{K^d} 2^{-(d+K^d+1)} K^{-d} = (2K)^{-d}/2. \tag{3.5}$$

It follows that for all $\beta \in \mathscr{B}$ and $N, L \in \mathbb{N}$ with $NL \geq J_1$, we have

$$|f_\beta(\boldsymbol{x}) - \phi_\beta(\boldsymbol{x})| \leq C_2 (NL)^{-\alpha(2/d+\rho/\alpha)}, \quad \text{for any } \boldsymbol{x} \in [0,1]^d \backslash \mathcal{H}. \tag{3.6}$$

For each index vector $\boldsymbol{\theta} \in \{1, 2, \cdots, K\}^d$ and any $\boldsymbol{x} \in \frac{1}{2} Q_{\boldsymbol{\theta}}$, where $\frac{1}{2} Q_{\boldsymbol{\theta}}$ denotes the cube whose sidelength is half of that of $Q_{\boldsymbol{\theta}}$ sharing the same center of $Q_{\boldsymbol{\theta}}$, since $Q_{\boldsymbol{\theta}}$ has a sidelength $\frac{1}{K}$, we have

$$|f_\beta(\boldsymbol{x})| = |g_{Q_{\boldsymbol{\theta}}}(\boldsymbol{x})| \geq |g_{Q_{\boldsymbol{\theta}}}(\boldsymbol{x}_{Q_{\boldsymbol{\theta}}})|/2 = \big(\tfrac{1}{2K}\big)^\alpha/4, \tag{3.7}$$

where $\boldsymbol{x}_{Q_{\boldsymbol{\theta}}}$ is the center of $Q_{\boldsymbol{\theta}}$. For fixed d, α, and ρ, there exists $J_2 > 0$ large enough such that, for any $N, L \in \mathbb{N}$ with $NL \geq J_2$, we have

$$\tfrac{1}{2^{2+\alpha}} \lfloor (NL)^{2/d+\rho/(2\alpha)} \rfloor^{-\alpha} > C_2 (NL)^{-\alpha(2/d+\rho/\alpha)}. \tag{3.8}$$

By (3.5), for any $\boldsymbol{\theta} \in \{1, 2, \cdots, K\}^d$, we have

$$\mu(\mathcal{H}) \leq (2K)^{-d}/2 < (2K)^{-d} = \mu(\tfrac{1}{2} Q_{\boldsymbol{\theta}}),$$

which means $\frac{1}{2} Q_{\boldsymbol{\theta}} \cap ([0,1]^d \backslash \mathcal{H})$ is not empty. Therefore, there exists $\boldsymbol{x}_{\boldsymbol{\theta}} \in \frac{1}{2} Q_{\boldsymbol{\theta}} \cap ([0,1]^d \backslash \mathcal{H})$ for each $\boldsymbol{\theta} \in \{1, 2, \cdots, K\}^d$ such that

$$|f_\beta(\boldsymbol{x}_{\boldsymbol{\theta}})| \geq \tfrac{1}{2^{2+\alpha}} \lfloor (NL)^{2/d+\rho/(2\alpha)} \rfloor^{-\alpha} > C_2 (NL)^{-\alpha(2/d+\rho/\alpha)} \geq |f_\beta(\boldsymbol{x}_{\boldsymbol{\theta}}) - \phi_\beta(\boldsymbol{x}_{\boldsymbol{\theta}})|,$$

where the first, the second, and the last inequalities come from (3.7), (3.8), and (3.6), respectively. In other words, for any $\beta \in \mathscr{B}$ and $\boldsymbol{\theta} \in \{1,2,\cdots,K\}^d$, $f_\beta(\boldsymbol{x}_{\boldsymbol{\theta}})$ and $\phi_\beta(\boldsymbol{x}_{\boldsymbol{\theta}})$ have the same sign. Then $\{\phi_\beta : \beta \in \mathscr{B}\}$ shatters $\{\boldsymbol{x}_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \{1,2,\cdots,K\}^d\}$ since $\{f_\beta : \beta \in \mathscr{B}\}$ shatters $\{\boldsymbol{x}_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \{1,2,\cdots,K\}^d\}$ as discussed in Step 1. Hence,

$$\mathrm{VCDim}\big(\{\phi_\beta : \beta \in \mathscr{B}\}\big) \geq K^d = b_\ell(N,L). \tag{3.9}$$

Step 3: Contradiction.

By (3.3) and (3.9), for any $N, L \in \mathbb{N}$ with $NL \geq J_0 = \max\{J_1, J_2\}$, we have

$$b_\ell(N,L) \leq \mathrm{VCDim}\big(\{\phi_\beta : \beta \in \mathscr{B}\}\big) \leq \mathrm{VCDim}\big(\mathrm{NN}(\#\text{input} = d;\ \text{maxwidth} \leq N;\ \#\text{layer} \leq L)\big) \leq b_u(N,L),$$

implying that

$$\lfloor (NL)^{2/d+\rho/(2\alpha)} \rfloor^d \leq C_1(LN+d+2)(N+1)L\ln\big((LN+d+2)(N+1)\big),$$

which is a contradiction for sufficiently large $N, L \in \mathbb{N}$. So we finish the proof. ∎

By Theorem 3.1, for any $\rho > 0$, the approximation rate cannot be $\mathcal{O}(N^{-(2\alpha/d+\rho)}L^{-(2/\alpha+\rho)})$, if we use FNNs in $\mathrm{NN}(\#\text{input} = d;\ \text{maxwidth} \leq N;\ \#\text{layer} \leq L)$ to approximate functions in $\mathrm{Lip}(\nu,\alpha,d)$. By a similar argument, we can show that the approximation rate cannot be $\mathcal{O}(N^{-2\alpha/d}L^{-(2/\alpha+\rho)})$ nor $\mathcal{O}(N^{-(2\alpha/d+\rho)}L^{-2\alpha/d})$. Hence, the approximation rate in Theorem 1.1 is nearly tight.

3.2. Tightness in the $L^{p,\infty}$-Norm. In Section 2, the quantitative approximation rate was obtained using the L^∞-norm on $[0,1]^d\backslash\mathcal{H}$ or the L^p-norm on $[0,1]^d$ for $p \in [1,\infty)$ as a consequence, while the L^∞-norm on $[0,1]^d$ was used to prove the unachievable approximation rate in Section 3.1. In order to make our results tight in the same norm, we introduce a new norm, the $L^{1,\infty}$-norm below. Similarly, we can define $L^{p,\infty}$-norm for $p \in [1,\infty)$. The main intuition of this $L^{1,\infty}$-norm is to rely on a small "don't-care" region, where the approximation error can be large, while uniformly controlling the approximation error in other "important" regions.

Definition 3.3. *Shrinking function. We define the shrinking function as*

$$\lambda(x) \coloneqq \tfrac{2^{-(2^x)}-2^{-(2^{x+1})}}{x}, \quad \text{for any } x \in [1,\infty).$$

It is clear that the shrinking function $\lambda(\cdot)$ is strictly decreasing in x on $[1,\infty)$ and satisfies

$$\sum_{j=J}^{\infty} j\lambda(j) \leq 2^{-(2^J)}, \text{ for any } J \in \mathbb{N}^+. \tag{3.10}$$

Actually, the shrinking function can be simply defined as $\lambda(x) \coloneqq \frac{2^{-x}}{x}$ for any $x \in [1,\infty)$, but the proof will be more complicated. Let $\lambda_{\mathrm{inv}}(\cdot)$ be the inverse function of $\lambda(\cdot)$, i.e., $\lambda(\lambda_{\mathrm{inv}}(x)) = x$ and $\lambda_{\mathrm{inv}}(\lambda(x)) = x$. Now we can define the "don't-care" region by using the shrinking function $\lambda(\cdot)$.

Definition 3.4. *Shrinking region. Let $\mathcal{S}(\ell) = \cup_{j=1}^{\ell}[j/\ell - \lambda(\ell), j/\ell]$. Then the shrinking region corresponding to integers k and d is defined as*

$$\Omega(k,d) \coloneqq \cup_{\ell=k}^{\infty}\Big(\cup_{i=1}^{d}\big\{\boldsymbol{x} = [x_1,\cdots,x_d]^T \in \mathbb{R}^d : x_i \in \mathcal{S}(\ell)\big\}\Big). \tag{3.11}$$

The shrinking region $\Omega(k,d)$ is the so-called "don't-care" region in this paper. Furthermore, $\Omega(k,d)$ shrinks gradually to a set of rational points in $[0,1]^d$ as $k \to \infty$, and the shrinking speed is determined by the shrinking function $\lambda(\cdot)$. More precisely,

$$\mu\big(\Omega(k,d)\big) \leq \sum_{j=k}^{\infty} dj\lambda(j) \leq d2^{-(2^k)}, \quad \text{for any } k,d \in \mathbb{N}^+. \tag{3.12}$$

Using the "don't-care" region, we are ready to define a new norm denoted as $L^{1,\infty}$.

DEFINITION 3.5. *For any $f \in L^1([0,1]^d) \cap L^\infty([0,1]^d)$, we define*

$$\|f\|_{L^{1,\infty}([0,1]^d)} := \sup\left\{\tfrac{1}{\ln k}\|f\|_{L^\infty([0,1]^d\setminus\Omega(k,d))} : k \geq 3,\ k \in \mathbb{N}\right\} + \|f\|_{L^1([0,1]^d)}.$$

It follows that $\mu(\Omega(k,d))$ decays quickly, so $\|f\|_{L^\infty([0,1]^d\setminus\Omega(k,d))}$ approaches to $\|f\|_{L^\infty([0,1]^d)}$ as $k \to \infty$. To define a norm weaker than the L^∞-norm, we need the factor $\frac{1}{\ln k}$ in Definition 3.5, which can be replaced by other functions going to 0 as $k \to +\infty$. It is easy to verify that $\|\cdot\|_{L^{1,\infty}([0,1]^d)}$ ($\|\cdot\|_{1,\infty}$ for short) is a norm satisfying

$$\|f\|_{L^1([0,1]^d)} \leq \|f\|_{1,\infty} \leq 2\|f\|_{L^\infty([0,1]^d)}, \quad \text{for any } f \in L^1([0,1]^d) \cap L^\infty([0,1]^d).$$

In fact, to make the $L^{1,\infty}$-norm well-defined, the shrinking function $\lambda(\cdot)$ can be any strictly decreasing function defined on $[1,\infty)$ with $\lim_{x\to\infty}\lambda(x) = 0$.

Theorem 3.6 below provides a quantitative approximation rate in the $L^{1,\infty}$-norm, which is asymptotically tight by Theorem 3.7.

THEOREM 3.6. *Let f be a continuous function on $[0,1]^d$. For any $L, N \in \mathbb{N}^+$, there exists a ReLU FNN ϕ with width $\max\{8d\lfloor N^{1/d}\rfloor + 4d, 12N + 14\}$ and depth $9L + 12$ such that*

$$\|f - \phi\|_{1,\infty} \leq 5\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d}).$$

Proof of Theorem 3.6. We divide the proof into two cases: the case $d = 1$ and the case $d \geq 2$.

Case 1: The case $d = 1$.

All the settings are the same as in Section 2.3 except the conditions on δ. The reader is referred to Section 2.3 for the definitions of notations. In Section 2.3, we require (2.1), but now we let $\delta = \delta(N, L, f) > 0$ be a sufficiently small number satisfying Equation (3.13), (3.14), and (3.15). It remains to estimate the approximation error in the $L^{1,\infty}$-norm. By Equation (2.7) and (2.8), we get

$$\begin{aligned}
\|f - \phi\|_{L^1([0,1])} &= \sum_{m=0}^{M-1}\sum_{\ell=0}^{L-1}\int_{Q_{m,\ell}} |f(x) - \phi(x)|dx + \int_{\mathcal{H}} |\bar{f}(x) - \bar{\phi}(x)|dx \\
&\leq \sum_{m=0}^{M-1}\sum_{\ell=0}^{L-1} \tfrac{1}{ML} 2\omega_f(\tfrac{1}{ML}) + 4ML\delta\omega_f(1) \\
&\leq 3\omega_f\big(\tfrac{1}{ML}\big),
\end{aligned}$$

where the last inequality comes from

$$4ML\delta\omega_f(1) \leq \omega_f(\tfrac{1}{ML}), \tag{3.13}$$

when δ is chosen to be small enough. In fact, we also require that δ satisfies

$$\tfrac{4\omega_f(1)}{\ln\lambda_{\text{inv}}(\delta)} \leq 2\omega_f(\tfrac{1}{ML}), \tag{3.14}$$

and

$$\tfrac{\lambda_{\text{inv}}(\delta)}{ML} \in \mathbb{N}^+, \tag{3.15}$$

where $\lambda(\cdot)$ and $\lambda_{\text{inv}}(\cdot)$ were introduced in Definition 3.3. By (3.14), we have

$$\sup\left\{\tfrac{\|f-\phi\|_{L^\infty([0,1]\setminus\Omega(k,d))}}{\ln k} : k \geq \lambda_{\text{inv}}(\delta),\ k \in \mathbb{N}\right\} \leq \sup\left\{\tfrac{4\omega_f(1)}{\ln k} : k \geq \lambda_{\text{inv}}(\delta),\ k \in \mathbb{N}\right\} \leq \tfrac{4\omega_f(1)}{\ln\lambda_{\text{inv}}(\delta)} \leq 2\omega_f(\tfrac{1}{ML}).$$

By (3.15) and the definition of $\Omega(k,d)$ in (3.11), if $k \le \lambda_{\text{inv}}(\delta)$, we have

$$\Omega(k,1) \supseteq \Omega(\lambda_{\text{inv}}(\delta),1) \supseteq \cup_{j=1}^{\lambda_{\text{inv}}(\delta)} \big[\tfrac{j}{\lambda_{\text{inv}}(\delta)} - \lambda(\lambda_{\text{inv}}(\delta)), \tfrac{j}{\lambda_{\text{inv}}(\delta)}\big] \supseteq \cup_{j=1}^{ML} \big[\tfrac{j}{ML} - \delta, \tfrac{j}{ML}\big] = \mathcal{H},$$

where the last equality comes from (2.2). Together with (2.8), it holds that

$$\sup\big\{\tfrac{1}{\ln k}\|f-\phi\|_{L^\infty([0,1]\setminus\Omega(k,1))} : 3 \le k \le \lambda_{\text{inv}}(\delta),\ k\in\mathbb{N}\big\} \le \tfrac{1}{\ln 3}\|f-\phi\|_{L^\infty([0,1]\setminus\mathcal{H})} \le 2\omega_f(\tfrac{1}{ML}).$$

Then we get

$$\sup\big\{\tfrac{1}{\ln k}\|f-\phi\|_{L^\infty([0,1]\setminus\Omega(k,1))} : k \ge 3,\ k\in\mathbb{N}\big\} \le 2\omega_f(\tfrac{1}{ML}).$$

Therefore, we have

$$\|f-\phi\|_{1,\infty} = \sup\big\{\tfrac{1}{\ln k}\|f-\phi\|_{L^\infty([0,1]\setminus\Omega(k,1))} : k \ge 3,\ k\in\mathbb{N}\big\} + \|f-\phi\|_{L^1([0,1])} \le 5\omega_f(\tfrac{1}{ML}) = 5\omega_f(N^{-2}L^{-2}).$$

So we finish the proof for the case $d=1$.

Case 2: The case $d \ge 2$.

All the settings are the same as in Section 2.4 except the conditions on δ. The reader is referred to Section 2.4 for the definitions of notations. In Section 2.4, we require (2.9), but now we let $\delta = \delta(N,L,d,f) > 0$ be a sufficiently small number satisfying Equation (3.16), (3.17), and (3.18). It remains to estimate the approximation error in the $L^{1,\infty}$-norm. By Equation (2.13) and (2.14), we have

$$\begin{aligned}
\|f-\phi\|_{L^1([0,1]^d)} &= \int_{\mathcal{H}} |\bar{f}-\bar{\phi}| d\boldsymbol{x} + \int_{[0,1]^d\setminus\mathcal{H}} |f-\phi| d\boldsymbol{x} \\
&\le \mu(\mathcal{H})\big(2\omega_f(\sqrt{d}) + 2\omega_f(\sqrt{d})\big) + \sum_{\boldsymbol{\theta}\in\{0,1,\cdots,n-1\}^d} \int_{Q_{\boldsymbol{\theta}}} |f-\phi| d\boldsymbol{x} \\
&\le 4J\delta d\omega_f(\sqrt{d}) + \sum_{\boldsymbol{\theta}\in\{0,1,\cdots,n-1\}^d} 2\omega_f(\tfrac{\sqrt{d}}{J})\mu(Q_{\boldsymbol{\theta}}) \\
&\le 3\omega_f(\tfrac{\sqrt{d}}{J}),
\end{aligned}$$

where the last inequality comes from

$$4J\delta d\omega_f(\sqrt{d}) \le \omega_f(\tfrac{\sqrt{d}}{J}). \tag{3.16}$$

In fact, we also require

$$\tfrac{4\omega_f(\sqrt{d})}{\ln \lambda_{\text{inv}}(\delta)} \le 2\omega_f(\tfrac{\sqrt{d}}{J}) \tag{3.17}$$

and

$$\tfrac{\lambda_{\text{inv}}(\delta)}{J} \in \mathbb{N}^+. \tag{3.18}$$

By Equation (3.17), we have

$$\sup\big\{\tfrac{\|f-\phi\|_{L^\infty([0,1]^d\setminus\Omega(k,d))}}{\ln k} : k \ge \lambda_{\text{inv}}(\delta),\ k\in\mathbb{N}\big\} \le \sup\big\{\tfrac{4\omega_f(\sqrt{d})}{\ln k} : k \ge \lambda_{\text{inv}}(\delta),\ k\in\mathbb{N}\big\} \le \tfrac{4\omega_f(\sqrt{d})}{\ln \lambda_{\text{inv}}(\delta)} \le 2\omega_f(\tfrac{\sqrt{d}}{J}).$$

By (3.18) and the definition of $\Omega(k,d)$, if $k \le \lambda_{\text{inv}}(\delta)$, we have

$$\Omega(k,d) \supseteq \Omega(\lambda_{\text{inv}}(\delta),d) \supseteq \cup_{i=1}^{d} \big\{\boldsymbol{x}\in[0,1]^d : x_i \in \cup_{j=1}^{J}\big[\tfrac{j}{J}-\delta,\ \tfrac{j}{J}\big]\big\} = \mathcal{H}.$$

Together with (2.14), it holds that

$$\sup\left\{\tfrac{1}{\ln k}\|f-\phi\|_{L^\infty([0,1]^d\setminus\Omega(k,d))} : 3\le k\le \lambda_{\text{inv}}(\delta),\ k\in\mathbb{N}\right\}\le \tfrac{1}{\ln 3}\|f-\phi\|_{L^\infty([0,1]^d\setminus\mathcal{H})}\le 2\omega_f(\tfrac{\sqrt{d}}{J}).$$

Furthermore,

$$\sup\left\{\tfrac{1}{\ln k}\|f-\phi\|_{L^\infty([0,1]^d\setminus\Omega(k,d))} : k\ge 3,\ k\in\mathbb{N}\right\}\le 2\omega_f(\tfrac{\sqrt{d}}{J}).$$

Therefore,

$$\|f-\phi\|_{1,\infty}=\sup\left\{\tfrac{\|f-\phi\|_{L^\infty([0,1]^d\setminus\Omega(k,d))}}{\ln k} : k\ge 3,\ k\in\mathbb{N}\right\}+\|f-\phi\|_{L^1([0,1])}\le 5\omega_f(\tfrac{\sqrt{d}}{J})\le 5\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d}),$$

where the last inequality comes from the fact $J=\lfloor N^{1/d}\rfloor^2\lfloor L^{2/d}\rfloor\ge\frac{N^{2/d}L^{2/d}}{8}$ for any $N,L\in\mathbb{N}^+$. So we finish the proof for the case $d\ge 2$. ∎

Note that $\omega_f(r)\le\nu r^\alpha$ for any $r\ge 0$ if $f\in\text{Lip}(\nu,\alpha,d)$. Hence, we have the quantitative approximation rate $5(8\sqrt{d})^\alpha\nu N^{-2\alpha/d}L^{-2\alpha/d}$ in the $L^{1,\infty}$-norm if we use ReLU FNNs with width $\max\{8d\lfloor N^{1/d}\rfloor+4d, 12N+14\}$ and depth $9L+12$ to approximate functions in $\text{Lip}(\nu,\alpha,d)$ on $[0,1]^d$.

The following theorem estimates an asymptotic and unachivable approximation rate in the $L^{1,\infty}$-norm, showing that the quantitative approximation rate in Theorem 3.6 is tight for the function class $\text{Lip}(\nu,\alpha,d)$.

THEOREM 3.7. *For any $\rho>0$ and $C>0$, there exists $f\in\text{Lip}(\nu,\alpha,d)$, for any $J_0=J_0(\rho,C,f)>0$, there exist $N,L\in\mathbb{N}$ with $NL\ge J_0$ such that*

$$\inf_{\phi\in\text{NN}(\#\text{input}=d;\,\text{maxwidth}\le N;\,\#\text{layer}\le L)}\|\phi-f\|_{1,\infty}\ge C\nu N^{-(2\alpha/d+\rho)}L^{-(2\alpha/d+\rho)}.$$

Proof of Theorem 3.7. The proof is mainly based on Theorem 3.1 and Equation (3.1). By Equation (3.12), let $\widetilde{\rho}=\rho/2$ and $\widetilde{J}_0\ge J_0+3$ be a large number satisfying

$$\tfrac{1}{\ln(NL)}(NL)^{-(2\alpha/d+\rho/2)}\ge(NL)^{-(2\alpha/d+\rho)},\quad\text{for any } N,L \text{ with } NL\ge\widetilde{J}_0,$$

and

$$\mu(\Omega(NL,d))\le d2^{-(2^{NL})}\le 2^{-(d+K^d)}K^{-d},\quad\text{for any } N,L \text{ with } NL\ge\widetilde{J}_0, \text{ where } K=\lfloor(NL)^{2/d+\widetilde{\rho}/(2\alpha)}\rfloor.$$

Set $\mathcal{H}=\Omega(NL,d)$ and denote $\text{NN}(\#\text{input}=d;\,\text{maxwidth}\le N;\,\#\text{layer}\le L)$ by $\mathcal{N}$ for short. Then by Theorem 3.1, there exist $N,L\in\mathbb{N}$ with $NL\ge\widetilde{J}_0\ge J_0+3$ such that

$$\inf_{\phi\in\mathcal{N}}\|\phi-f\|_{L^\infty([0,1]^d\setminus\Omega(NL,d))}\ge C\nu(NL)^{-(2\alpha/d+\widetilde{\rho})}=C\nu(NL)^{-(2\alpha/d+\rho/2)}.$$

It follows that

$$\begin{aligned}\inf_{\phi\in\mathcal{N}}\|\phi-f\|_{1,\infty}&\ge\inf_{\phi\in\mathcal{N}}\sup\left\{\tfrac{1}{\ln k}\|\phi-f\|_{L^\infty([0,1]^d\setminus\Omega(k,d))} : k\ge 3,\ k\in\mathbb{N}\right\}\\&\ge\inf_{\phi\in\mathcal{N}}\tfrac{1}{\ln(NL)}\|\phi-f\|_{L^\infty([0,1]^d\setminus\Omega(NL,d))}\\&\ge C\nu(NL)^{-(2\alpha/d+\rho)}.\end{aligned}$$

So, we finish the proof. ∎

4. Neural Networks in Practice. This section is concerned with neural networks in practice, e.g., approximating functions defined on irregular domains or domains with a low-dimensional structure, and neural network computation in parallel computing.

4.1. Approximation on Irregular Domain. In this section, we consider approximating continuous functions defined on irregular domains by deep ReLU FNNs. The construction is through extending the target function to a cubic domain, applying Theorem 1.1, and finally restricting the constructed FNN back to the irregular domain.

Given any uniformly continuous and real-valued function f defined on a metric space S with a metric $d_S(\cdot,\cdot)$, we define the (optimal) modulus of continuity of f on a subset $E \subseteq S$ as

$$\omega_f^E(r) \coloneqq \sup\{|f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)| : \boldsymbol{x}_1, \boldsymbol{x}_2 \in E,\, d_S(\boldsymbol{x}_1, \boldsymbol{x}_2) \le r\}, \quad \text{for any } r \ge 0.$$

For the purpose of consistency and simplicity, $\omega_f(\cdot)$ is short of $\omega_f^{[0,1]^d}(\cdot)$.

First, let us present two lemmas for (approximately) extending (almost) continuous functions on E to (almost) continuous functions on S. These lemmas are similar to the well-known results for extending Lipchitz or differentiable functions in [43, 59]. We generalize these results to a broader class of functions required in the proof of Theorem 4.3.

LEMMA 4.1 (Approximate Extension of Almost-Continuous Functions). *Assume S is a metric space with a metric $d_S(\cdot,\cdot)$ and $\omega : [0,\infty) \to [0,\infty)$ is an increasing function with*

$$\omega(r_1 + r_2) \le \omega(r_1) + \omega(r_2), \quad \text{for any } r_1, r_2 \in [0,\infty). \tag{4.1}$$

Let the real-valued function f be defined on a subset $E \subseteq S$ and satisfy

$$|f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)| \le \omega(d_S(\boldsymbol{x}_1, \boldsymbol{x}_2) + \Delta), \quad \text{for any } \boldsymbol{x}_1, \boldsymbol{x}_2 \in E, \tag{4.2}$$

where Δ is a positive constant independent of f. Then there exists a function g defined on S such that

$$0 \le f(\boldsymbol{x}) - g(\boldsymbol{x}) \le \omega(\Delta), \quad \text{for any } \boldsymbol{x} \in E$$

and

$$|g(\boldsymbol{x}_1) - g(\boldsymbol{x}_2)| \le \omega(d_S(\boldsymbol{x}_1, \boldsymbol{x}_2)), \quad \text{for any } \boldsymbol{x}_1, \boldsymbol{x}_2 \in S.$$

In Lemma 4.1, g is an approximate extension of f defined on E to a new domain S with an approximation error $\omega(\Delta)$. In a special case when $\Delta = 0$ and $\omega(0) = 0$, g is an exact extension of f.

Proof of Lemma 4.1. Define

$$g(\boldsymbol{x}) \coloneqq \sup_{\boldsymbol{z} \in E} \big(f(\boldsymbol{z}) - \omega(d_S(\boldsymbol{z}, \boldsymbol{x}) + \Delta)\big).$$

By Equation (4.2), we have $f(\boldsymbol{x}_1) - \omega(d_S(\boldsymbol{x}_1, \boldsymbol{x}_2) + \Delta) \le f(\boldsymbol{x}_2)$ for any $\boldsymbol{x}_1, \boldsymbol{x}_2 \in E$. It holds that $g(\boldsymbol{x}) \le f(\boldsymbol{x})$ for any $\boldsymbol{x} \in E$. Together with

$$g(\boldsymbol{x}) = \sup_{z \in E} \big(f(\boldsymbol{z}) - \omega(d_S(\boldsymbol{z}, \boldsymbol{x}) + \Delta)\big) \ge f(\boldsymbol{x}) - \omega(d_S(\boldsymbol{x}, \boldsymbol{x}) + \Delta) = f(\boldsymbol{x}) - \omega(\Delta), \quad \text{for any } \boldsymbol{x} \in E,$$

it follows that $0 \le f(\boldsymbol{x}) - g(\boldsymbol{x}) \le \omega(\Delta)$ for any $\boldsymbol{x} \in E$. By Equation (4.1) and the fact

$$\sup_{\boldsymbol{z} \in E} f_1(\boldsymbol{z}) - \sup_{\boldsymbol{z} \in E} f_2(\boldsymbol{z}) \le \sup_{\boldsymbol{z} \in E} \big(f_1(\boldsymbol{z}) - f_2(\boldsymbol{z})\big), \quad \text{for any functions } f_1, f_2,$$

we have

$$\begin{aligned} g(\boldsymbol{x}_1) - g(\boldsymbol{x}_2) &= \sup_{\boldsymbol{z}\in E}\big(f(\boldsymbol{z}) - \omega(d_S(\boldsymbol{z},\boldsymbol{x}_1))\big) - \sup_{\boldsymbol{z}\in E}\big(f(\boldsymbol{z}) - \omega(d_S(\boldsymbol{z},\boldsymbol{x}_2))\big) \\ &\le \sup_{\boldsymbol{z}\in E}\big(\omega(d_S(\boldsymbol{z},\boldsymbol{x}_1)) - \omega(d_S(\boldsymbol{z},\boldsymbol{x}_2))\big) \\ &\le \sup_{\boldsymbol{z}\in E}\omega\big(d_S(\boldsymbol{z},\boldsymbol{x}_1) - d_S(\boldsymbol{z},\boldsymbol{x}_2)\big) \\ &\le \sup_{\boldsymbol{z}\in E}\omega(d_S(\boldsymbol{x}_1,\boldsymbol{x}_2)) \\ &= \omega(d_S(\boldsymbol{x}_1,\boldsymbol{x}_2)), \end{aligned}$$

for any $\boldsymbol{x}_1, \boldsymbol{x}_2 \in S$. Similarly, we have $g(\boldsymbol{x}_2) - g(\boldsymbol{x}_1) \le \omega(d_S(\boldsymbol{x}_1,\boldsymbol{x}_2))$, which implies

$$|g(\boldsymbol{x}_1) - g(\boldsymbol{x}_2)| \le \omega(d_S(\boldsymbol{x}_1,\boldsymbol{x}_2)).$$

So we finish the proof. ∎

Next, we introduce a lemma below for extending continuous functions defined on $E \subseteq S$ to continuous functions defined on S preserving the modulus of continuity.

LEMMA 4.2 (Extension of Continuous Functions). *Suppose f is a uniformly continuous function defined on a subset $E \subseteq S$, where S is a metric space with a metric $d_S(\cdot,\cdot)$, then there exists a uniformly continuous function g on S such that $f(x) = g(x)$ for $x \in E$ and $\omega_f^E(r) = \omega_g^S(r)$ for any $r \ge 0$.*

Proof of Lemma 4.2. By the application of Lemma 4.1 with $\omega(r) = \omega_f^E(r)$ for $r \ge 0$ and $\Delta = 0$, we know that there exists $g : S \to \mathbb{R}$ such that

$$0 \le f(\boldsymbol{x}) - g(\boldsymbol{x}) \le \omega_f^E(\Delta) = 0, \quad \text{for any } \boldsymbol{x} \in E,$$

and

$$|g(\boldsymbol{x}_1) - g(\boldsymbol{x}_2)| \le \omega_f^E(d_S(\boldsymbol{x}_1,\boldsymbol{x}_2)), \quad \text{for any } \boldsymbol{x}_1, \boldsymbol{x}_2 \in S.$$

The equation above and the uniform continuity of f imply that g is uniformly continuous. It also follows that

$$f(\boldsymbol{x}) = g(\boldsymbol{x}), \quad \text{for any } \boldsymbol{x} \in E, \quad \text{and} \quad \omega_g^S(r) \le \omega_f^E(r), \quad \text{for any } r \ge 0,$$

since $\omega_g^S(\cdot)$ is the optimal modulus of continuity of g. Note that $\omega_f^E(\cdot)$ is the optimal moduls of continuity of f and

$$|f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)| = |g(\boldsymbol{x}_1) - g(\boldsymbol{x}_2)| \le \omega_g^S(d_S(\boldsymbol{x}_1,\boldsymbol{x}_2)), \quad \text{for any } \boldsymbol{x}_1, \boldsymbol{x}_2 \in E.$$

Hence, $\omega_f^E(r) \le \omega_g^S(r)$ for all $r \ge 0$, which implies $\omega_f^E(r) = \omega_g^S(r)$ since we have proved that $\omega_g^S(r) \le \omega_f^E(r)$ for all $r \ge 0$. So we finish the proof. ∎

Now we are ready to introduce and prove the main theorem of this section, which extends Theorem 1.1 to an irregular domain as follows.

THEOREM 4.3. *Let f be a uniformly continuous function defined on $E \subseteq [-R,R]^d$. For any arbitrary $L \in \mathbb{N}^+$, $N \in \mathbb{N}^+$, and $\eta > 0$, there exists a ReLU FNN ϕ with width $\max\{8d\lfloor N^{1/d}\rfloor + 4d, 12N + 14\}$ and depth $9L + 12$ such that*

$$\|f - \phi\|_{L^\infty(E\setminus\mathcal{H})} \le 5\omega_f^E(16R\sqrt{d}N^{-2/d}L^{-2/d}), \quad \text{and} \quad \|\phi\|_{L^\infty(E)} \le \|f\|_{L^\infty(E)} + 2\omega_f^E(2R\sqrt{d}),$$

where $\mathcal{H} \subseteq E$ with $\mu(\mathcal{H}) \le \eta$.

As in Corollary 1.2, it is an immediate result to replace the $L^\infty(E\backslash\mathcal{H})$-norm with the $L^p(E)$-norm for $p\in[1,\infty)$. Now let us prove Theorem 4.3.

Proof of Theorem 4.3. By Lemma 4.2, f can be extended to $\mathbb{R}^d$ such that

$$\omega_f^{\mathbb{R}^d}(r)=\omega_f^E(r),\quad \text{for any } r\geq 0.$$

Define

$$\tilde{f}(\boldsymbol{x})\coloneqq f(2R\boldsymbol{x}-R),\quad \text{for any } \boldsymbol{x}\in\mathbb{R}^d.$$

It follows that

$$\omega_{\tilde{f}}^{\mathbb{R}^d}(r)=\omega_f^{\mathbb{R}^d}(2Rr)=\omega_f^E(2Rr),\quad \text{for any } r\geq 0. \tag{4.3}$$

By Theorem 1.1, there exists a ReLU FNN $\tilde{\phi}$ with width $\max\{8d\lfloor N^{1/d}\rfloor+4d, 12N+14\}$ and depth $9L+12$ such that

$$\|\tilde{f}-\tilde{\phi}\|_{L^\infty([0,1]^d\backslash\widetilde{\mathcal{H}})}\leq 5\omega_{\tilde{f}}^{\mathbb{R}^d}(8\sqrt{d}N^{-2/d}L^{-2/d}),\quad \text{and}\quad \|\tilde{\phi}\|_{L^\infty([0,1]^d)}\leq|\tilde{f}(0)|+\omega_{\tilde{f}}^{\mathbb{R}^d}(\sqrt{d}),$$

for any $\widetilde{\mathcal{H}}\in[0,1]^d$ with $\mu(\widetilde{\mathcal{H}})\leq\frac{\eta}{(2R)^d}$. Define $\mathcal{H}_0=2R\widetilde{\mathcal{H}}-R\subseteq[-R,R]^d$ and

$$\phi(\boldsymbol{x})\coloneqq\tilde{\phi}(\tfrac{1}{2R}\boldsymbol{x}+\tfrac{1}{2}),\quad \text{for any } \boldsymbol{x}\in\mathbb{R}^d.$$

Then for any $\boldsymbol{x}\in[-R,R]^d\backslash\mathcal{H}_0$, we have

$$\begin{aligned}|f(\boldsymbol{x})-\phi(\boldsymbol{x})|&=|\tilde{f}(\tfrac{1}{2R}\boldsymbol{x}+\tfrac{1}{2})-\tilde{\phi}(\tfrac{1}{2R}\boldsymbol{x}+\tfrac{1}{2})|\\&\leq\|\tilde{f}-\tilde{\phi}\|_{L^\infty([0,1]^d\backslash\widetilde{\mathcal{H}})}\\&\leq 5\omega_{\tilde{f}}^{\mathbb{R}^d}(8\sqrt{d}N^{-2/d}L^{-2/d})\\&=5\omega_f^E(16R\sqrt{d}N^{-2/d}L^{-2/d}).\end{aligned}$$

Set $\mathcal{H}=E\cap\mathcal{H}_0$, then $\mu(\mathcal{H})\leq\mu(\mathcal{H}_0)=(2R)^d\mu(\widetilde{\mathcal{H}})\leq\eta$ and $E\backslash\mathcal{H}\subseteq[-R,R]^d\backslash\mathcal{H}_0$, which implies

$$\|f-\phi\|_{L^\infty(E\backslash\mathcal{H})}\leq 5\omega_f^E(16R\sqrt{d}N^{-2/d}L^{-2/d}).$$

Besides,

$$\begin{aligned}\|\phi\|_{L^\infty(E)}\leq\|\phi\|_{L^\infty([-R,R]^d)}=\|\tilde{\phi}\|_{L^\infty([0,1]^d)}\leq|\tilde{f}(0)|+\omega_{\tilde{f}}^{\mathbb{R}^d}(\sqrt{d})&\leq|\tilde{f}(\tfrac{1}{2R}\boldsymbol{x}_E+\tfrac{1}{2})|+2\omega_{\tilde{f}}^{\mathbb{R}^d}(\sqrt{d})\\&=|f(\boldsymbol{x}_E)|+2\omega_f^E(2R\sqrt{d})\\&=\|f\|_{L^\infty(E)}+2\omega_f^E(2R\sqrt{d}),\end{aligned}$$

where $\boldsymbol{x}_E$ is a point in E. So we finish the proof. ∎

4.2. Approximation in a Neighborhood of a Low-Dimensional Manifold. In this section, we study neural network approximation of functions defined in a neighborhood of a low-dimensional manifold and prove Theorem 1.3 in this setting. Let us first introduce Theorem 4.4 and Lemma 4.5 which are required to prove Theorem 1.3.

THEOREM 4.4 (Theorem 3.1 of [3]). *Let $\mathcal{M}$ be a compact $d_{\mathcal{M}}$-dimensional Riemannian submanifold of $\mathbb{R}^d$ having condition number $1/\tau$, volume V, and geodesic covering regularity $\mathcal{R}$. Fix $\delta\in(0,1)$ and $\gamma\in(0,1)$. Let $\boldsymbol{A}=\sqrt{\frac{d}{d_\delta}}\Phi$, where $\Phi\in\mathbb{R}^{d_\delta\times d}$ is a random orthoprojector with*

$$d_\delta=\mathcal{O}\Big(\frac{d_{\mathcal{M}}\ln(dV\mathcal{R}\tau^{-1}\delta^{-1})\ln(1/\gamma)}{\delta^2}\Big).$$

If $d_\delta \le d$, then with probability at least $1-\gamma$, the following statement holds: For every $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{M}$,

$$(1-\delta)|\boldsymbol{x}_1 - \boldsymbol{x}_2| \le |\boldsymbol{A}\boldsymbol{x}_1 - \boldsymbol{A}\boldsymbol{x}_2| \le (1+\delta)|\boldsymbol{x}_1 - \boldsymbol{x}_2|.$$

Theorem 4.4 shows the existence of a linear projector $\boldsymbol{A} \in \mathbb{R}^{d_\delta \times d}$ that maps a low-dimensional manifold in a high-dimensional space to a low-dimensional space nearly preserving distance. With this projection $\boldsymbol{A}$ available, we can prove Theorem 1.3 via constructing a ReLU FNN defined in the low-dimensional space using Theorem 4.3 and hence the curse of dimensionality is lessened. The ideas of the proof are summarized in the following Table 1.

In Table 1 and the detailed proof later, we introduce a new notation $\mathcal{SL}(E)$ for any compact set $E \subseteq \mathbb{R}^d$ as the "smallest" element of E. Specifically, $\mathcal{SL}(E)$ is defined as the unique point in $\cap_{k=1}^d E_k$, where

$$E_k \coloneqq \{\boldsymbol{x} \in E_{k-1} : x_k = s_k\}, \quad s_k \coloneqq \inf\{x_k : [x_1, x_2, \cdots, x_d]^T \in E_{k-1}\}, \quad \text{for } k = 1, 2, \cdots, d,$$

and $E_0 = E$. The compactness of E ensures that $\cap_{k=1}^d E_k$ is in fact one point belonging to E. The introduction of $\mathcal{SL}(\cdot)$ uniquely formulates a low-dimensional function $\tilde{f}$ representing a high-dimensional function f defined on $\mathcal{M}_\varepsilon$ by

$$\tilde{f}(\boldsymbol{y}) \coloneqq f(\boldsymbol{x}_{\boldsymbol{y}}), \quad \text{where } \boldsymbol{x}_{\boldsymbol{y}} = \mathcal{SL}\big(\{\boldsymbol{x} \in \mathcal{M}_\varepsilon : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}\}\big), \quad \text{for any } \boldsymbol{y} \in \boldsymbol{A}(\mathcal{M}_\varepsilon) \subseteq \mathbb{R}^{d_\delta}.$$

As we shall see later, $\tilde{f}$ can approximate f well because $\{\boldsymbol{x} \in \mathcal{M}_\varepsilon : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}\}$ is contained in a small ball of radius $\mathcal{O}(\varepsilon)$ for any $\boldsymbol{y} \in \boldsymbol{A}(\mathcal{M}_\varepsilon)$. There are many other alternative ways to define $\mathcal{SL}(\cdot)$ as long as the definition ensures that $\mathcal{SL}(E)$ contains only one element. For example, $\mathcal{SL}(E)$ can be defined as any arbitrary point in E. For another example, $y \in \boldsymbol{A}(\mathcal{M})$ cannot guarantee $\boldsymbol{x}_{\boldsymbol{y}} = \mathcal{SL}\big(\{\boldsymbol{x} \in \mathcal{M}_\varepsilon : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}\}\big) \in \mathcal{M}$ in the current definition, but in practice we can choose $\mathcal{SL}\big(\{\boldsymbol{x} \in \mathcal{M} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}\}\big)$ as $\boldsymbol{x}_{\boldsymbol{y}}$ to ensure that $\boldsymbol{x}_{\boldsymbol{y}} \in \mathcal{M}$, which might be beneficial for potential applications.

TABLE 1

Main steps of the proof of Theorem 1.3. Step 1: dimension reduction via the nearly isometric projection operator $\boldsymbol{A}$ provided by Theorem 4.4 to obtain an "equivalent" function $\tilde{f}$ of f in a low-dimensional domain using $\boldsymbol{x}_{\boldsymbol{y}} = \mathcal{SL}(\{\boldsymbol{x} \in \mathcal{M}_\varepsilon : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}\})$. Step 2: construct a ReLU FNN $\tilde{\phi} \approx \tilde{f}$ by Theorem 4.3. Step 3: define a ReLU FNN ϕ in the original high-dimensional domain via the projection $\boldsymbol{A}$. Step 4: verify that the approximation error of $\phi \approx f$ satisfies our requirement.

$f(\boldsymbol{x})$ for $\boldsymbol{x} \in \mathcal{M}_\varepsilon \subseteq [0,1]^d$	Step 4 $\approx$	$\phi(\boldsymbol{x}) \coloneqq \tilde{\phi}(\boldsymbol{A}\boldsymbol{x})$ for $\boldsymbol{x} \in \mathcal{M}_\varepsilon \subseteq [0,1]^d$
Step 1 $\Big\Updownarrow$ $\boldsymbol{x}_{\boldsymbol{y}} = \mathcal{SL}\big(\{\boldsymbol{x} \in \mathcal{M}_\varepsilon : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}\}\big)$		Step 3 $\Big\Updownarrow$ $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$
$\tilde{f}(\boldsymbol{y}) \coloneqq f(\boldsymbol{x}_{\boldsymbol{y}})$ for $\boldsymbol{y} \in \boldsymbol{A}(\mathcal{M}_\varepsilon) \subseteq \mathbb{R}^{d_\delta}$	Step 2 $\approx$	$\tilde{\phi}(\boldsymbol{y})$ for $\boldsymbol{y} \in \boldsymbol{A}(\mathcal{M}_\varepsilon) \subseteq \mathbb{R}^{d_\delta}$

Note that in Step 2 in Table 1, the approximation $\tilde{f} \approx \tilde{\phi}$ is only valid in the L^∞-norm outside a "don't-care" region $\mathcal{H}$ with an arbitrarily small measure $\mu(\mathcal{H})$. Hence, when we verify the approximation error in Step 4, it is important to ensure that $\mu\big(\{\boldsymbol{x} \in \mathcal{M}_\varepsilon : \boldsymbol{A}\boldsymbol{x} \in \mathcal{H}\}\big)$ is well-controlled. Hence, Lemma 4.5 below is introduced to characterize the behavior of the Lebesgue measure under a linear transform. For any set $E \subseteq \mathbb{R}^d$ and a matrix $\boldsymbol{A} \in \mathbb{R}^{d' \times d}$, we use the notation $\boldsymbol{A}(E) \coloneqq \{\boldsymbol{A}\boldsymbol{x} : \boldsymbol{x} \in E\}$.

LEMMA 4.5. *Given $R > 0$ and $d', d \in \mathbb{N}^+$ with $d' \le d$, let $\boldsymbol{A} \in \mathbb{R}^{d' \times d}$ be a row independent matrix, then*

$$\mu\big(\{\boldsymbol{x} \in \mathbb{R}^d : \boldsymbol{A}\boldsymbol{x} \in \mathcal{H}\} \cap [-R, R]^d\big) \to 0 \quad \text{if} \quad \mu(\mathcal{H}) \to 0,$$

for any Lebesgue measurable set $\mathcal{H} \subseteq \mathbb{R}^{d'}$.

Proof of Lemma 4.5. Since $\boldsymbol{A} \in \mathbb{R}^{d' \times d}$ is row independent, there exists $\boldsymbol{A}_0 \in \mathbb{R}^{(d-d') \times d}$ such that $\widetilde{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{A}_0 \end{bmatrix}$ is nonsingular, which implies

$$\boldsymbol{A} = \boldsymbol{I}_{d'} \boldsymbol{A} = [\boldsymbol{I}_{d'}, \boldsymbol{0}] \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{A}_0 \end{bmatrix} = [\boldsymbol{I}_{d'}, \boldsymbol{0}] \widetilde{\boldsymbol{A}}.$$

Define

$$\widetilde{\mathcal{H}} \coloneqq \{\boldsymbol{x} \in \mathbb{R}^d : [\boldsymbol{I}_{d'}, \boldsymbol{0}]\boldsymbol{x} \in \mathcal{H}\} = \{\boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in \mathbb{R}^d : [x_1, \cdots, x_{d'}]^T \in \mathcal{H}\} = \mathcal{H} \otimes \mathbb{R}^{d-d'}.$$

It follows that

$$\begin{aligned} \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \in \mathcal{H}\} \cap [-R, R]^d &= \{\boldsymbol{x} : [\boldsymbol{I}_{d'}, \boldsymbol{0}]\widetilde{\boldsymbol{A}}\boldsymbol{x} \in \mathcal{H}\} \cap [-R, R]^d \\ &= \widetilde{\boldsymbol{A}}^{-1}\big(\{\boldsymbol{y} : [\boldsymbol{I}_{d'}, \boldsymbol{0}]\boldsymbol{y} \in \mathcal{H}\} \cap \widetilde{\boldsymbol{A}}([-R, R]^d)\big) \\ &= \widetilde{\boldsymbol{A}}^{-1}\big(\widetilde{\mathcal{H}} \cap \widetilde{\boldsymbol{A}}([-R, R]^d)\big). \end{aligned}$$

There exists $R' > 0$ such that $\widetilde{\boldsymbol{A}}([-R, R]^d) \subseteq [-R', R']^d$, then

$$\widetilde{\mathcal{H}} \cap \widetilde{\boldsymbol{A}}([-R, R]^d) \subseteq \widetilde{\mathcal{H}} \cap [-R', R']^d = \big(\mathcal{H} \otimes \mathbb{R}^{d-d'}\big) \cap [-R', R']^d \subseteq \mathcal{H} \otimes [-R', R']^{d-d'},$$

which means

$$\mu\big(\widetilde{\mathcal{H}} \cap \widetilde{\boldsymbol{A}}([-R, R]^d)\big) \to 0 \quad \text{if} \quad \mu(\mathcal{H}) \to 0.$$

Since $\widetilde{\boldsymbol{A}}^{-1}$ is nonsingular,

$$\mu\Big(\widetilde{\boldsymbol{A}}^{-1}\big(\widetilde{\mathcal{H}} \cap \widetilde{\boldsymbol{A}}([-R, R]^d)\big)\Big) \to 0 \quad \text{if} \quad \mu\big(\widetilde{\mathcal{H}} \cap \widetilde{\boldsymbol{A}}([-R, R]^d)\big) \to 0.$$

Therefore, we have

$$\mu\big(\{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \in \mathcal{H}\} \cap [-R, R]^d\big) \to 0 \quad \text{if} \quad \mu(\mathcal{H}) \to 0.$$

So we finish the proof. ∎

Now we are ready to prove Theorem 1.3.

Proof of Theorem 1.3. By Theorem 4.4, there exists a matrix $\boldsymbol{A} \in \mathbb{R}^{d_\delta \times d}$ such that

$$\boldsymbol{A}\boldsymbol{A}^T = \tfrac{d}{d_\delta} I_{d_\delta}, \tag{4.4}$$

where $\boldsymbol{I}_{d_\delta}$ is an identity matrix of size $d_\delta \times d_\delta$, and

$$(1-\delta)|\boldsymbol{x}_1 - \boldsymbol{x}_2| \le |\boldsymbol{A}\boldsymbol{x}_1 - \boldsymbol{A}\boldsymbol{x}_2| \le (1+\delta)|\boldsymbol{x}_1 - \boldsymbol{x}_2|, \quad \text{for any } \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{M}. \tag{4.5}$$

Given any $\boldsymbol{y} \in \boldsymbol{A}(\mathcal{M}_\varepsilon)$, then $\{\boldsymbol{x} \in \mathcal{M}_\varepsilon : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}\}$ is a nonzero compact set. Let $\boldsymbol{x}_{\boldsymbol{y}} = \mathcal{SL}\big(\{\boldsymbol{x} \in \mathcal{M}_\varepsilon : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}\}\big)$, then we define $\tilde{f}$ on $\boldsymbol{A}(\mathcal{M}_\varepsilon)$ as $\tilde{f}(\boldsymbol{y}) = f(\boldsymbol{x}_{\boldsymbol{y}})$.

For any $\boldsymbol{y}_1, \boldsymbol{y}_2 \in \boldsymbol{A}(\mathcal{M}_\varepsilon)$, let $\boldsymbol{x}_i = \mathcal{SL}\big(\{\boldsymbol{x} \in \mathcal{M}_\varepsilon : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}_i\}\big)$, then $\boldsymbol{x}_i \in \mathcal{M}_\varepsilon$ for $i = 1, 2$. By the definition of $\mathcal{M}_\varepsilon$, there exist $\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2 \in \mathcal{M}$ such that $|\tilde{\boldsymbol{x}}_i - \boldsymbol{x}_i| \le \varepsilon$ for $i = 1, 2$. It follows that

$$|\tilde{f}(\boldsymbol{y}_1) - \tilde{f}(\boldsymbol{y}_2)| = |f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)| \le \omega_f(|\boldsymbol{x}_1 - \boldsymbol{x}_2|) \le \omega_f(|\tilde{\boldsymbol{x}}_1 - \tilde{\boldsymbol{x}}_2| + 2\varepsilon) \le \omega_f\big(\tfrac{1}{1-\delta}|\boldsymbol{A}\tilde{\boldsymbol{x}}_1 - \boldsymbol{A}\tilde{\boldsymbol{x}}_2| + 2\varepsilon\big),$$

where the last inequality comes from (4.5). By the triangular inequality, we have

$$\begin{aligned}|\tilde{f}(\boldsymbol{y}_1)-\tilde{f}(\boldsymbol{y}_2)| &\le \omega_f\big(\tfrac{1}{1-\delta}|\boldsymbol{A}\boldsymbol{x}_1-\boldsymbol{A}\boldsymbol{x}_2|+\tfrac{1}{1-\delta}|\boldsymbol{A}\boldsymbol{x}_1-\boldsymbol{A}\tilde{\boldsymbol{x}}_1|+\tfrac{1}{1-\delta}|\boldsymbol{A}\boldsymbol{x}_2-\boldsymbol{A}\tilde{\boldsymbol{x}}_2|+2\varepsilon\big)\\ &\le \omega_f\big(\tfrac{1}{1-\delta}|\boldsymbol{A}\boldsymbol{x}_1-\boldsymbol{A}\boldsymbol{x}_2|+\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}}+2\varepsilon\big)\\ &\le \omega_f\big(\tfrac{1}{1-\delta}|\boldsymbol{y}_1-\boldsymbol{y}_2|+\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}}+2\varepsilon\big).\end{aligned}$$

Set $\omega(r)=\omega_f(\frac{1}{1-\delta}r)$ for any $r\ge 0$ and $\Delta=2\varepsilon\sqrt{\frac{d}{d_\delta}}+2\varepsilon(1-\delta)$, then

$$|\tilde{f}(\boldsymbol{y}_1)-\tilde{f}(\boldsymbol{y}_2)|\le\omega(|\boldsymbol{y}_1-\boldsymbol{y}_2|+\Delta),\quad \text{for any } \boldsymbol{y}_1,\boldsymbol{y}_2\in\boldsymbol{A}(\mathcal{M}_\varepsilon)\subseteq\mathbb{R}^{d_\delta}.$$

By Lemma 4.1, there exists $\tilde{g}$ defined on $\mathbb{R}^{d_\delta}$ such that

$$|\tilde{g}(\boldsymbol{y})-\tilde{f}(\boldsymbol{y})|\le\omega(\Delta)=\omega_f\big(\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}}+2\varepsilon\big),\quad \text{for any } \boldsymbol{y}\in\boldsymbol{A}(\mathcal{M}_\varepsilon), \tag{4.6}$$

and

$$|\tilde{g}(\boldsymbol{y}_1)-\tilde{g}(\boldsymbol{y}_2)|\le\omega(|\boldsymbol{y}_1-\boldsymbol{y}_2|)=\omega_f\big(\tfrac{1}{1-\delta}|\boldsymbol{y}_1-\boldsymbol{y}_2|\big),\quad \text{for any } \boldsymbol{y}_1,\boldsymbol{y}_2\in\mathbb{R}^{d_\delta}.$$

It follows that

$$\omega_{\tilde{g}}^{\mathbb{R}^{d_\delta}}(r)\le\omega_f\big(\tfrac{r}{1-\delta}\big),\quad \text{for any } r\ge 0. \tag{4.7}$$

By Equation (4.4) and the definition of $\mathcal{M}_\varepsilon$ in Equation (1.2), it is easy to check that

$$\boldsymbol{A}(\mathcal{M}_\varepsilon)\subseteq\boldsymbol{A}([0,1]^d)\subseteq[-\sqrt{\tfrac{d}{d_\delta}},\sqrt{\tfrac{d}{d_\delta}}]^{d_\delta}.$$

By the application of Theorem 4.3 with $E=[-\sqrt{\frac{d}{d_\delta}},\sqrt{\frac{d}{d_\delta}}]^{d_\delta}$, there exists a ReLU FNN $\tilde{\phi}$ with width $\max\big\{8d_\delta\lfloor N^{1/d_\delta}\rfloor+4d_\delta, 12N+14\big\}$ and depth $9L+12$ such that

$$\|\tilde{g}-\tilde{\phi}\|_{L^\infty(E\setminus\widetilde{\mathcal{H}})}\le 5\omega_{\tilde{g}}^{E}\big(\tfrac{16d}{\sqrt{d_\delta}}N^{-2/d_\delta}L^{-2/d_\delta}\big)\quad\text{and}\quad\|\tilde{\phi}\|_{L^\infty(E)}\le\|\tilde{g}\|_{L^\infty(E)}+2\omega_{\tilde{g}}^{E}\big(\tfrac{2d}{\sqrt{d_\delta}}\big), \tag{4.8}$$

where $\widetilde{\mathcal{H}}\subseteq E$ with $\mu(\widetilde{\mathcal{H}})\le\eta$, and $\eta>0$ is a small number to be determined later.

Define $\phi\coloneqq\tilde{\phi}\circ\boldsymbol{A}$, i.e., $\phi(\boldsymbol{x})\coloneqq\tilde{\phi}(\boldsymbol{A}\boldsymbol{x})$ for any $\boldsymbol{x}\in\mathbb{R}^d$. Then ϕ is also a ReLU FNN with width $\max\big\{8d_\delta\lfloor N^{1/d_\delta}\rfloor+4d_\delta, 12N+14\big\}$ and depth $9L+12$. Let $\mathcal{H}\coloneqq\{\boldsymbol{x}:\boldsymbol{A}\boldsymbol{x}\in\widetilde{\mathcal{H}}\}\cap\mathcal{M}_\varepsilon$. By Lemma 4.5,

$$\eta\to 0\Longrightarrow\mu(\widetilde{\mathcal{H}})\to 0\Longrightarrow\mu(\mathcal{H})\to 0\Longrightarrow\mu_\varrho(\mathcal{H})\to 0. \tag{4.9}$$

Therefore,

$$\begin{aligned}\|f-\phi\|_{L^p([0,1]^d,\mu_\varrho)}&=\bigg(\int_{\mathcal{M}_\varepsilon}|f(\boldsymbol{x})-\phi(\boldsymbol{x})|^p d\mu_\varrho(\boldsymbol{x})\bigg)^{1/p}\\ &\le\bigg(\int_{\mathcal{M}_\varepsilon\setminus\mathcal{H}}|f(\boldsymbol{x})-\phi(\boldsymbol{x})|^p d\mu_\varrho(\boldsymbol{x})\bigg)^{1/p}+\bigg(\int_{\mathcal{H}}|f(\boldsymbol{x})-\phi(\boldsymbol{x})|^p d\mu_\varrho(\boldsymbol{x})\bigg)^{1/p}\coloneqq\mathscr{S}_1+\mathscr{S}_2.\end{aligned}$$

First, let us estimate $\mathscr{S}_1$. For any $\boldsymbol{x}\in\mathcal{M}_\varepsilon\setminus\mathcal{H}$, set $\boldsymbol{y}=\boldsymbol{A}\boldsymbol{x}$ and $\boldsymbol{x}_{\boldsymbol{y}}=\mathcal{SL}\big(\{\boldsymbol{z}\in\mathbb{R}^d:\boldsymbol{A}\boldsymbol{z}=\boldsymbol{y}\}\big)$, there exist $\tilde{\boldsymbol{x}},\tilde{\boldsymbol{x}}_{\boldsymbol{y}}\in\mathcal{M}$ such that $|\tilde{\boldsymbol{x}}-\boldsymbol{x}|\le\varepsilon$ and $|\tilde{\boldsymbol{x}}_{\boldsymbol{y}}-\boldsymbol{x}_{\boldsymbol{y}}|\le\varepsilon$. It follows that

$$\begin{aligned}|\boldsymbol{x}-\boldsymbol{x}_{\boldsymbol{y}}|&\le|\tilde{\boldsymbol{x}}-\tilde{\boldsymbol{x}}_{\boldsymbol{y}}|+2\varepsilon\le\tfrac{1}{1-\delta}|\boldsymbol{A}\tilde{\boldsymbol{x}}-\boldsymbol{A}\tilde{\boldsymbol{x}}_{\boldsymbol{y}}|+2\varepsilon\\ &\le\tfrac{1}{1-\delta}\big(|\boldsymbol{A}\tilde{\boldsymbol{x}}-\boldsymbol{A}\boldsymbol{x}|+|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{A}\boldsymbol{x}_{\boldsymbol{y}}|+|\boldsymbol{A}\boldsymbol{x}_{\boldsymbol{y}}-\boldsymbol{A}\tilde{\boldsymbol{x}}_{\boldsymbol{y}}|\big)+2\varepsilon\\ &=\tfrac{1}{1-\delta}\big(|\boldsymbol{A}\tilde{\boldsymbol{x}}-\boldsymbol{A}\boldsymbol{x}|+|\boldsymbol{A}\boldsymbol{x}_{\boldsymbol{y}}-\boldsymbol{A}\tilde{\boldsymbol{x}}_{\boldsymbol{y}}|\big)+2\varepsilon\\ &\le\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}}+2\varepsilon.\end{aligned} \tag{4.10}$$

In fact, the above equation implies that $\{\boldsymbol{x} \in \mathcal{M}_\varepsilon : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}\}$ is contained in a small ball of radius $\mathcal{O}(\varepsilon)$ for any $\boldsymbol{y} \in \boldsymbol{A}(\mathcal{M}_\varepsilon)$ as we mentioned previously.

Together with Equation (4.6), we have

$$\begin{aligned}
|f(\boldsymbol{x}) - \phi(\boldsymbol{x})| &\le |f(\boldsymbol{x}) - f(\boldsymbol{x}_{\boldsymbol{y}})| + |f(\boldsymbol{x}_{\boldsymbol{y}}) - \phi(\boldsymbol{x})| \\
&\le \omega_f\Big(\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}} + 2\varepsilon\Big) + |\tilde{f}(\boldsymbol{y}) - \tilde{\phi}(\boldsymbol{y})| \\
&\le \omega_f\Big(\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}} + 2\varepsilon\Big) + |\tilde{f}(\boldsymbol{y}) - \tilde{g}(\boldsymbol{y})| + |\tilde{g}(\boldsymbol{y}) - \tilde{\phi}(\boldsymbol{y})| \\
&\le \omega_f\Big(\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}} + 2\varepsilon\Big) + \omega_f\Big(\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}} + 2\varepsilon\Big) + 5\omega_{\tilde{g}}^E\big(\tfrac{16d}{\sqrt{d_\delta}} N^{-2/d_\delta} L^{-2/d_\delta}\big) \\
&\le 2\omega_f\Big(\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}} + 2\varepsilon\Big) + 5\omega_f\big(\tfrac{16d}{(1-\delta)\sqrt{d_\delta}} N^{-2/d_\delta} L^{-2/d_\delta}\big).
\end{aligned}$$

It follows that

$$\mathscr{S}_1 \le 2\omega_f\Big(\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}} + 2\varepsilon\Big) + 5\omega_f\Big(\tfrac{16d}{(1-\delta)\sqrt{d_\delta}} N^{-2/d_\delta} L^{-2/d_\delta}\Big).$$

Then we focus on $\mathscr{S}_2$. By Equation (4.8) and the definition of $\phi = \tilde{\phi} \circ \boldsymbol{A}$,

$$\begin{aligned}
\|f - \phi\|_{L^\infty(\mathcal{M}_\varepsilon)} &\le \|f\|_{L^\infty(\mathcal{M}_\varepsilon)} + \|\phi\|_{L^\infty(\mathcal{M}_\varepsilon)} \\
&= \|f\|_{L^\infty(\mathcal{M}_\varepsilon)} + \|\tilde{\phi}\|_{L^\infty(\boldsymbol{A}(\mathcal{M}_\varepsilon))} \\
&\le \|f\|_{L^\infty(\mathcal{M}_\varepsilon)} + \|\tilde{g}\|_{L^\infty(E)} + 2\omega_{\tilde{g}}^E\big(\tfrac{2d}{\sqrt{d_\delta}}\big) \\
&\le \|f\|_{L^\infty(\mathcal{M}_\varepsilon)} + \|\tilde{g}\|_{L^\infty(E)} + 2\omega_f\big(\tfrac{2d}{(1-\delta)\sqrt{d_\delta}}\big).
\end{aligned}$$

Hence,

$$\mathscr{S}_2 \le \left(\Big(\|f\|_{L^\infty(\mathcal{M}_\varepsilon)} + \|\tilde{g}\|_{L^\infty(E)} + 2\omega_f\big(\tfrac{2d}{(1-\delta)\sqrt{d_\delta}}\big)\Big)^p \mu_\varrho(\mathcal{H})\right)^{1/p}.$$

Together with Equation (4.9), we know

$$\eta \to 0 \Longrightarrow \mu_\varrho(\mathcal{H}) \to 0 \Longrightarrow \mathscr{S}_2 \to 0.$$

So we can choose a sufficiently small $\eta > 0$ such that

$$\mathscr{S}_2 \le \omega_f\Big(\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}} + 2\varepsilon\Big).$$

Therefore,

$$\begin{aligned}
\|f - \phi\|_{L^p([0,1]^d,\mu_\varrho)} = \mathscr{S}_1 + \mathscr{S}_2 &\le 3\omega_f\Big(\tfrac{2\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}} + 2\varepsilon\Big) + 5\omega_f\Big(\tfrac{16d}{(1-\delta)\sqrt{d_\delta}} N^{-2/d_\delta} L^{-2/d_\delta}\Big) \\
&\le 3\omega_f\Big(\tfrac{4\varepsilon}{1-\delta}\sqrt{\tfrac{d}{d_\delta}}\Big) + 5\omega_f\Big(\tfrac{16d}{(1-\delta)\sqrt{d_\delta}} N^{-2/d_\delta} L^{-2/d_\delta}\Big).
\end{aligned}$$

Hence, we have finished the proof of this theorem. ∎

It is worth emphasizing that the approximation error $\mathcal{O}\Big(\omega_f\big(\mathcal{O}(\varepsilon)\big) + \omega_f\big(\mathcal{O}(N^{-2/d_\delta} L^{-2/d_\delta})\big)\Big)$ in Theorem 1.3 is equal to $\mathcal{O}\Big(\omega_f\big(\mathcal{O}(N^{-2/d_\delta} L^{-2/d_\delta})\big)\Big)$ when $\varepsilon \le N^{-2/d_\delta} L^{-2/d_\delta}$.

The application of Theorem 4.4 and the proof of Theorem 1.3 in fact inspire an efficient two-step algorithm for high-dimensional learning problems: in the first step, high-dimensional data are projected to a low-dimensional space via a random projection; in the second step, a deep learning algorithm is applied to learn from the low-dimensional data. By Theorem 4.4 and 1.3, the deep learning algorithm in the low-dimensional space can still provide good results with a high probability.

4.3. Optimal ReLU FNN Structure in Parallel Computing. In this section, we show how to select the best ReLU FNN to approximate functions in $\mathrm{Lip}(\nu,\alpha,d)$ on a d-dimensional cube, if the approximation error ε and the number of parallel computing cores (processors) p are given. We choose the best ReLU FNN by minimizing the time complexity in each training iteration. The analysis in this section is valid up to a constant prefactor.

Assume $\phi_{\boldsymbol{\theta}} \in \mathrm{NN}(\#\mathrm{input}=d;\ \mathrm{widthvec}=[N]^L)$, $N, L \in \mathbb{N}^+$, where $\boldsymbol{\theta}$ is the vector including all parameters of $\phi_{\boldsymbol{\theta}}$. By the basic knowledge of parallel computing (see [33] for more details), we have the following Table 2.

TABLE 2
Time complexity of one training iteration for an FNN of width N and depth L.

Number of cores p	Time Complexity	
	Evaluating $\phi_{\boldsymbol{\theta}}(\boldsymbol{x})$	Evaluating $\frac{\partial \phi_{\boldsymbol{\theta}}(\boldsymbol{x})}{\partial \boldsymbol{\theta}}$
$p \in [1, N]$	$\mathcal{O}(N^2L/p)$	$\mathcal{O}(N^2L/p)$
$p \in (N, N^2]$	$\mathcal{O}\big(L(N^2/p + \ln \frac{p}{N})\big)$	$\mathcal{O}\big(L(N^2/p + \ln \frac{p}{N})\big)$
$p \in (N^2, \infty)$	$\mathcal{O}(L \ln N)$	$\mathcal{O}(L \ln N)$

For the sake of simplicity, we assume that the training batch size is $\mathcal{O}(1)$. Denote the time complexity of each training iteration as $T(n, L)$, then

$$T(N,L) = \begin{cases} \mathcal{O}(N^2L/p), & p \in [1, N], \\ \mathcal{O}\big(L(N^2/p + \ln \frac{p}{N})\big), & p \in (N, N^2], \\ \mathcal{O}(L \ln N), & p \in (N^2, \infty). \end{cases}$$

Theorem 3.6 and 3.7 imply that the approximation error ε is essentially $\mathcal{O}((NL)^{-2\alpha/d})$. Hence, we can get the optimal size of ReLU FNNs via the optimization problem below:

$$\begin{aligned} (N_{\mathrm{opt}}, L_{\mathrm{opt}}) &= \arg\min_{N,\ L}\ T(N,L) \\ \text{subject to } &\begin{cases} \varepsilon = \mathcal{O}\big((NL)^{-2\alpha/d}\big), \\ N, L, p \in N^+. \end{cases} \end{aligned} \tag{4.11}$$

To simplify the discussion, we have the following assumptions:

- Dropping the notation $\mathcal{O}(\cdot)$ sometimes while assuming asymptotic analysis with the abuse of notations.
- N, L, and p are allowed to be real numbers.
- Replacing $\mathcal{O}\big((NL)^{-(2\alpha/d+\rho)}\big) \le \varepsilon \le \mathcal{O}\big((NL)^{-2\alpha/d}\big)$ with $\varepsilon = (NL)^{-2\alpha/d}$ since $\rho > 0$ is arbitrary.

With $\varepsilon = (NL)^{-2\alpha/d}$, we have

$$\begin{aligned} \overline{T}(N,L) &:= \begin{cases} N^2L/p & p \in [1, N], \\ L(N^2/p + \ln \frac{p}{N}), & p \in (N, N^2], \\ L(1 + \ln N), & p \in [N^2, \infty), \end{cases} \\ &= \begin{cases} N\varepsilon^{-d/(2\alpha)}/p, & N \in [p, \infty), \\ N\varepsilon^{-d/(2\alpha)}/p + \frac{1}{N}\varepsilon^{-d/(2\alpha)} \ln \frac{p}{N}, & N \in [\sqrt{p}, p), \\ \frac{1+\ln N}{N}\varepsilon^{-d/(2\alpha)}, & N \in [1, \sqrt{p}). \end{cases} \end{aligned} \tag{4.12}$$

Then we get $T(N,L) = \mathcal{O}(\overline{T}(N,L))$. Therefore, the optimization problem in (4.11) can be simplified to

$$\begin{aligned} (N_{\mathrm{opt}}, L_{\mathrm{opt}}) &= \arg\min_{N,\ L}\ \overline{T}(N,L) \\ \text{subject to } &\begin{cases} \varepsilon = (NL)^{-2\alpha/d}, \\ N, L, p \in [1, \infty). \end{cases} \end{aligned} \tag{4.13}$$

By (4.12), $\overline{T}(N,L)$ is independent of L on the condition that $\varepsilon = (NL)^{-2\alpha/d}$. Therefore, we may denote $\overline{T}(N,L)$ by $\overline{T}(N)$. Now we consider two cases: the case $p = \mathcal{O}(1)$ and the case $p \gg \mathcal{O}(1)$.

Case 1: The case $p = \mathcal{O}(1)$.

It is clear that $\overline{T}(N)$ is increasing in N when $N \in [p,\infty)$ by (4.12). Together with $p = \mathcal{O}(1)$, then $\mathcal{O}(\sqrt{p}) = \mathcal{O}(p) = \mathcal{O}(1)$. Therefore, $N_{\text{opt}} = \mathcal{O}(1)$ and $L_{\text{opt}} = \mathcal{O}(\varepsilon^{-d/(2\alpha)})$. Note that we regard d as a constant ($\mathcal{O}(1)$) in above analysis, N_{opt} should be $\mathcal{O}(d)$ in fact.

Case 2: The case $p \gg \mathcal{O}(1)$.

Since $\varepsilon = (NL)^{-2\alpha/d}$, we have $N \le \varepsilon^{-d/(2\alpha)}$. We only need to consider the monotonicity of $\overline{T}(N)$ on $[1,\varepsilon^{-d/(2\alpha)}]$. Together with (4.12), this case can be divided into two sub-cases: the sub-case $\sqrt{p} \le \varepsilon^{-d/(2\alpha)}$ and the sub-case $\sqrt{p} > \varepsilon^{-d/(2\alpha)}$.

Case 2.1: The sub-case $\sqrt{p} > \varepsilon^{-d/(2\alpha)}$.

$\sqrt{p} > \varepsilon^{-d/(2\alpha)}$ implies $[1,\varepsilon^{-d/(2\alpha)}] \subseteq [1,\sqrt{q}]$. Hence, $\overline{T}(N)$ is decreasing in N on $[1,\varepsilon^{-d/(2\alpha)}]$. It follows that $N_{\text{opt}} = \mathcal{O}(\varepsilon^{-d/(2\alpha)})$ and that $L_{\text{opt}} = \mathcal{O}(1)$.

Case 2.2: The sub-case $\sqrt{p} \le \varepsilon^{-d/(2\alpha)}$.

For this sub-case, N_{opt} and N_{opt} are hard to estimate. However, we can give a rough range of N_{opt}. Since $\overline{T}(N)$ is decreasing in N on $[1,\sqrt{p}]$ and increasing in N on $[p,\infty)$, the minimum of $\overline{T}(N)$ is achieved on $[\sqrt{p},p]$. Hence, $N_{\text{opt}} \in [\mathcal{O}(\sqrt{p}),\mathcal{O}(p)] \cap [\mathcal{O}(\sqrt{p}),\mathcal{O}(\varepsilon^{-d/(2\alpha)})]$ and $L_{\text{opt}} = \mathcal{O}(\varepsilon^{-d/(2\alpha)}/N_{\text{opt}})$.

5. Conclusion and future work. This paper aims at a quantitative and optimal approximation rate of ReLU FNNs in terms of both width and depth simultaneously to approximate continuous functions. It was shown that ReLU FNNs with width $\mathcal{O}(N)$ and depth $\mathcal{O}(L)$ can approximate an arbitrary continuous function on a d-dimensional cube with an approximation rate $5\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d})$. In particular, when f is a Hölder continuous function of order α with a Lipschitz constant ν, the approximation rate is $5(8\sqrt{d})^{\alpha}\nu N^{-2\alpha/d}L^{-2\alpha/d}$ and it is asymptotically tight. We also extended our analysis to the case when the domain of f is irregular and showed the same approximation rate. In practical applications, it is usually believed that real data are sampled from an ε-neighborhood of a $d_{\mathcal{M}}$-dimensional smooth manifold $\mathcal{M} \subseteq [0,1]^d$ with $d_{\mathcal{M}} \ll d$. In the case of an essentially low-dimensional domain, we show an approximation rate $3\omega_f\big(\frac{4\varepsilon}{1-\delta}\sqrt{\frac{d}{d_\delta}}\big) + 5\omega_f\big(\frac{16d}{(1-\delta)\sqrt{d_\delta}}N^{-2/d_\delta}L^{-2/d_\delta}\big)$ for ReLU FNNs to approximate f in the ε-neighborhood, $d_\delta = \mathcal{O}\big(d_{\mathcal{M}}\frac{\ln(d/\delta)}{\delta^2}\big)$ for any given $\delta \in (0,1)$.

Besides, we studied how to select the best ReLU FNN to approximate continuous function in parallel computing. In particular, ReLU FNNs with depth $\mathcal{O}(1)$ are the best choices if the number of parallel computing cores p is sufficiently large. ReLU FNNs with width $\mathcal{O}(d)$ are best choices if $p = \mathcal{O}(1)$. The width of best ReLU FNNs is between $\mathcal{O}(\sqrt{p})$ and $\mathcal{O}(p)$ if p is moderate.

We would like to remark that our analysis was based on the fully connected feed-forward neural networks and the ReLU activation function. It would be very interesting to generalize our conclusions to neural networks with other types of architectures (e.g., convolutional neural networks) and activation functions (e.g., tanh and sigmoid functions). Another important direction is to extend our results to the L^∞-norm on the whole domain without the "don't-care" region. Besides, if identity maps are allowed in the construction of neural networks as in the residual networks [26], the size of FNNs in our construction can be further optimized. Finally, the proposed analysis could be generalized to other function spaces with explicit formulas to characterize the approximation error. These will be left as future work.

Acknowledgments. H. Yang thanks the support of the start-up grant by the Department of Mathematics at the National University of Singapore, the support of the Ministry of Education in Singapore under the grant MOE2018-T2-2-147.

REFERENCES

[1] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, *Convolutional neural networks for speech recognition*, IEEE/ACM Transactions on Audio, Speech, and Language Processing, 22 (2014), pp. 1533–1545, https://doi.org/10.1109/TASLP.2014.2339736.

[2] M. Anthony and P. L. Bartlett, *Neural Network Learning: Theoretical Foundations*, Cambridge University Press, New York, NY, USA, 1st ed., 2009.

[3] R. G. Baraniuk and M. B. Wakin, *Random projections of smooth manifolds*, Foundations of Computational Mathematics, 9 (2009), pp. 51–77, https://doi.org/10.1007/s10208-007-9011-z, https://doi.org/10.1007/s10208-007-9011-z.

[4] A. R. Barron, *Universal approximation bounds for superpositions of a sigmoidal function*, IEEE Transactions on Information Theory, 39 (1993), pp. 930–945, https://doi.org/10.1109/18.256500.

[5] P. Bartlett, V. Maiorov, and R. Meir, *Almost linear VC dimension bounds for piecewise polynomial networks*, Neural Computation, 10 (1998), pp. 217–3.

[6] M. Bianchini and F. Scarselli, *On the complexity of neural network classifiers: A comparison between shallow and deep architectures*, IEEE Transactions on Neural Networks and Learning Systems, 25 (2014), pp. 1553–1565, https://doi.org/10.1109/TNNLS.2013.2293637.

[7] E. K. Blum and L. K. Li, *Approximation theory and feedforward networks*, Neural Networks, 4 (1991), pp. 511 – 515, https://doi.org/https://doi.org/10.1016/0893-6080(91)90047-9, http://www.sciencedirect.com/science/article/pii/0893608091900479.

[8] D. S. Broomhead and D. Lowe, *Multivariable Functional Interpolation and Adaptive Networks*, Complex Systems 2, (1988), pp. 321–355.

[9] J. Cai, D. Li, J. Sun, and K. Wang, *Enhanced expressive power and fast training of neural networks by random projections*, CoRR, abs/1811.09054 (2018), http://arxiv.org/abs/1811.09054, https://arxiv.org/abs/1811.09054.

[10] S. Chen and D. Donoho, *Basis pursuit*, in Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers, vol. 1, Oct 1994, pp. 41–44 vol.1, https://doi.org/10.1109/ACSSC.1994.471413.

[11] C. K. Chui, S.-B. Lin, and D.-X. Zhou, *Construction of neural networks for realization of localized deep learning*.

[12] D. C. Cireşan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, *Flexible, high performance convolutional neural networks for image classification*, in Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two, IJCAI'11, AAAI Press, 2011, pp. 1237–1242, https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-210, http://dx.doi.org/10.5591/978-1-57735-516-8/IJCAI11-210.

[13] D. Costarelli and A. R. Sambucini, *Saturation classes for max-product neural network operators activated by sigmoidal functions*, Results in Mathematics, 72 (2017), pp. 1555 – 1569, https://doi.org/10.1007/s00025-017-0692-6.

[14] D. Costarelli and G. Vinti, *Convergence for a family of neural network operators in orlicz spaces*, Mathematische Nachrichten, 290 (2017), pp. 226–235, https://onlinelibrary.wiley.com/doi/abs/10.1002/mana.201600006.

[15] D. Costarelli and G. Vinti, *Approximation results in orlicz spaces for sequences of kantorovich max-product neural network operators*, Results in Mathematics, 73 (2018), pp. 1 – 15, https://doi.org/10.1007/s00025-018-0799-4.

[16] G. Cybenko, *Approximation by superpositions of a sigmoidal function*, MCSS, 2 (1989), pp. 303–314.

[17] I. Daubechies, R. DeVore, S. Foucart, B. Hanin, and G. Petrova, *Nonlinear approximation and (deep) relu networks*, 2019.

[18] R. DEVORE and A. RON, *Approximation using scattered shifts of a multivariate function*, Transactions of the American Mathematical Society, 362 (2010), pp. 6205–6229, http://www.jstor.org/stable/40997201.

[19] R. A. DeVore, *Nonlinear approximation*, Acta Numerica, 7 (1998), p. 51–150, https://doi.org/10.1017/S0962492900002816.

[20] W. E, J. Han, and A. Jentzen, *Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations*, Communications in Mathematics and Statistics, 5 (2017), pp. 349–380, https://doi.org/10.1007/s40304-017-0117-6, https://doi.org/10.1007/s40304-017-0117-6.

[21] W. E and Q. Wang, *Exponential convergence of the deep neural network approximation for analytic functions*, CoRR, abs/1807.00297 (2018), http://arxiv.org/abs/1807.00297, https://arxiv.org/abs/1807.00297.

[22] J. Han, A. Jentzen, and W. E, *Solving high-dimensional partial differential equations using deep learning*, Proceedings of the National Academy of Sciences, 115 (2018), pp. 8505–8510, https://doi.org/10.1073/pnas.1718942115, https://www.pnas.org/content/115/34/8505, https://arxiv.org/abs/https://www.pnas.org/content/115/34/8505.full.pdf.

[23] T. Hangelbroek and A. Ron, *Nonlinear approximation using gaussian kernels*, Journal of Functional Analysis, 259 (2010), pp. 203 – 219, https://doi.org/https://doi.org/10.1016/j.jfa.2010.02.001, http://www.sciencedirect.com/science/article/pii/S0022123610000467.

[24] B. Hanin and M. Sellke, *Approximating continuous functions by ReLU nets of minimal width*, (2017), https://arxiv.org/abs/1710.11278.

[25] N. Harvey, C. Liaw, and A. Mehrabian, *Nearly-tight VC-dimension bounds for piecewise linear neural networks*, in Proceedings of the 2017 Conference on Learning Theory, S. Kale and O. Shamir, eds., vol. 65 of Proceedings of Machine Learning Research, Amsterdam, Netherlands, 07–10 Jul 2017, PMLR, pp. 1064–1068, http://proceedings.mlr.press/v65/harvey17a.html.

[26] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016, pp. 770–778, https://doi.org/10.1109/CVPR.2016.90.

[27] K. Hornik, *Approximation capabilities of multilayer feedforward networks*, Neural Networks, 4 (1991), pp. 251

– 257, https://doi.org/https://doi.org/10.1016/0893-6080(91)90009-T, http://www.sciencedirect.com/science/article/pii/089360809190009T.

[28] K. Hornik, M. Stinchcombe, and H. White, *Multilayer feedforward networks are universal approximators*, Neural Networks, 2 (1989), pp. 359 – 366, https://doi.org/https://doi.org/10.1016/0893-6080(89)90020-8, http://www.sciencedirect.com/science/article/pii/0893608089900208.

[29] K. Kawaguchi, *Deep learning without poor local minima*, in Advances in Neural Information Processing Systems 29, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, eds., Curran Associates, Inc., 2016, pp. 586–594, http://papers.nips.cc/paper/6112-deep-learning-without-poor-local-minima.pdf.

[30] K. Kawaguchi and Y. Bengio, *Depth with nonlinearity creates no bad local minima in resnets*, (2018), https://arxiv.org/abs/1810.09038, https://arxiv.org/abs/1810.09038.

[31] M. J. Kearns and R. E. Schapire, *Efficient distribution-free learning of probabilistic concepts*, J. Comput. Syst. Sci., 48 (1994), pp. 464–497, https://doi.org/10.1016/S0022-0000(05)80062-5, http://dx.doi.org/10.1016/S0022-0000(05)80062-5.

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, *Imagenet classification with deep convolutional neural networks*, in Advances in Neural Information Processing Systems 25, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds., Curran Associates, Inc., 2012, pp. 1097–1105, http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

[33] V. Kumar, *Introduction to Parallel Computing*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd ed., 2002.

[34] V. Kůrková, *Kolmogorov's theorem and multilayer neural networks*, Neural Networks, 5 (1992), pp. 501 – 506, https://doi.org/https://doi.org/10.1016/0893-6080(92)90012-8, http://www.sciencedirect.com/science/article/pii/0893608092900128.

[35] G. Lewicki and G. Marino, *Approximation of functions of finite variation by superpositions of a sigmoidal function*, Applied Mathematics Letters, 17 (2004), pp. 1147 – 1152, http://www.sciencedirect.com/science/article/pii/S089396590481694X.

[36] S. Liang and R. Srikant, *Why deep neural networks?*, CoRR, abs/1610.04161 (2016), http://arxiv.org/abs/1610.04161, https://arxiv.org/abs/1610.04161.

[37] S. Lin, X. Liu, Y. Rong, and Z. Xu, *Almost optimal estimates for approximation and learning by radial basis function networks*, Machine Learning, 95 (2014), pp. 147–164, https://doi.org/10.1007/s10994-013-5406-z, https://doi.org/10.1007/s10994-013-5406-z.

[38] B. Llanas and F. Sainz, *Constructive approximate interpolation by neural networks*, Journal of Computational and Applied Mathematics, 188 (2006), pp. 283 – 308, http://www.sciencedirect.com/science/article/pii/S0377042705002566.

[39] Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang, *The expressive power of neural networks: A view from the width*, in Advances in Neural Information Processing Systems 30, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds., Curran Associates, Inc., 2017, pp. 6231–6239, http://papers.nips.cc/paper/7203-the-expressive-power-of-neural-networks-a-view-from-the-width.pdf.

[40] Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang, *The expressive power of neural networks: A view from the width*, CoRR, abs/1709.02540 (2017), http://arxiv.org/abs/1709.02540, https://arxiv.org/abs/1709.02540.

[41] V. Maiorov and A. Pinkus, *Lower bounds for approximation by mlp neural networks*, Neurocomputing, 25 (1999), pp. 81 – 91, https://doi.org/https://doi.org/10.1016/S0925-2312(98)00111-8, http://www.sciencedirect.com/science/article/pii/S0925231298001118.

[42] S. G. Mallat and Z. Zhang, *Matching pursuits with time-frequency dictionaries*, IEEE Transactions on Signal Processing, 41 (1993), pp. 3397–3415, https://doi.org/10.1109/78.258082.

[43] E. J. McShane, *Extension of range of functions*, Bull. Amer. Math. Soc., 40 (1934), pp. 837–842, https://projecteuclid.org:443/euclid.bams/1183497871.

[44] H. Montanelli and Q. Du, *New error bounds for deep networks using sparse grids*, (2017), https://arxiv.org/abs/1712.08688.

[45] H. Montanelli, H. Yang, and Q. Du, *Deep relu networks overcome the curse of dimensionality for bandlimited functions*, (2019), https://arxiv.org/abs/1903.00735v1, https://arxiv.org/abs/1903.00735.

[46] G. F. Montufar, R. Pascanu, K. Cho, and Y. Bengio, *On the number of linear regions of deep neural networks*, in Advances in Neural Information Processing Systems 27, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds., Curran Associates, Inc., 2014, pp. 2924–2932, http://papers.nips.cc/paper/5422-on-the-number-of-linear-regions-of-deep-neural-networks.pdf.

[47] Q. N. Nguyen and M. Hein, *The loss surface of deep and wide neural networks*, CoRR, abs/1704.08045 (2017), http://arxiv.org/abs/1704.08045, https://arxiv.org/abs/1704.08045.

[48] J. Park and I. W. Sandberg, *Universal approximation using radial-basis-function networks*, Neural Computation, 3 (1991), pp. 246–257, https://doi.org/10.1162/neco.1991.3.2.246.

[49] P. Petersen and F. Voigtlaender, *Optimal approximation of piecewise smooth functions using deep ReLU neural networks*, Neural Networks, 108 (2018), pp. 296 – 330, https://doi.org/https://doi.org/10.1016/j.neunet.2018.08.019, http://www.sciencedirect.com/science/article/pii/S0893608018302454.

[50] P. Petrushev, *Multivariate n-term rational and piecewise polynomial approximation*, Journal of Approximation Theory, 121 (2003), pp. 158 – 197, https://doi.org/https://doi.org/10.1016/S0021-9045(02)00060-6, http://www.sciencedirect.

com/science/article/pii/S0021904502000606.

[51] D. Rolnick and M. Tegmark, *The power of deeper networks for expressing natural functions*, CoRR, abs/1705.05502 (2017), http://arxiv.org/abs/1705.05502, https://arxiv.org/abs/1705.05502.

[52] I. Safran and O. Shamir, *Depth-width tradeoffs in approximating natural functions with neural networks*, in Proceedings of the 34th International Conference on Machine Learning, D. Precup and Y. W. Teh, eds., vol. 70 of Proceedings of Machine Learning Research, International Convention Centre, Sydney, Australia, 06–11 Aug 2017, PMLR, pp. 2979–2987, http://proceedings.mlr.press/v70/safran17a.html.

[53] A. Sakurai, *Tight bounds for the VC-dimension of piecewise polynomial networks*, in Advances in Neural Information Processing Systems, Neural information processing systems foundation, 1999, pp. 323–329.

[54] D. Scherer, A. Müller, and S. Behnke, *Evaluation of pooling operations in convolutional architectures for object recognition*, in Artificial Neural Networks – ICANN 2010, K. Diamantaras, W. Duch, and L. S. Iliadis, eds., Berlin, Heidelberg, 2010, Springer Berlin Heidelberg, pp. 92–101.

[55] J. Schmidt-Hieber, *Nonparametric regression using deep neural networks with ReLU activation function*, (2017), https://arxiv.org/abs/1708.06633.

[56] U. Shaham, A. Cloninger, and R. R. Coifman, *Provable approximation properties for deep neural networks*, Applied and Computational Harmonic Analysis, 44 (2018), pp. 537 – 557, https://doi.org/https://doi.org/10.1016/j.acha.2016.04.003, http://www.sciencedirect.com/science/article/pii/S1063520316300033.

[57] Z. Shen, H. Yang, and S. Zhang, *Nonlinear Approximation via Compositions*, arXiv e-prints, (2019), arXiv:1902.10170, p. arXiv:1902.10170, https://arxiv.org/abs/1902.10170.

[58] T. Suzuki, *Adaptivity of deep reLU network for learning in besov and mixed smooth besov spaces: optimal rate and curse of dimensionality*, in International Conference on Learning Representations, 2019, https://openreview.net/forum?id=H1ebTsActm.

[59] H. Whitney, *Analytic extensions of differentiable functions defined in closed sets*, Transactions of the American Mathematical Society, 36 (1934), pp. 63–89, http://www.jstor.org/stable/1989708.

[60] T. F. Xie and F. L. Cao, *The rate of approximation of gaussian radial basis neural networks in continuous function space*, Acta Mathematica Sinica, English Series, 29 (2013), pp. 295–302, https://doi.org/10.1007/s10114-012-1369-4, https://doi.org/10.1007/s10114-012-1369-4.

[61] D. Yarotsky, *Error bounds for approximations with deep ReLU networks*, Neural Networks, 94 (2017), pp. 103 – 114, https://doi.org/https://doi.org/10.1016/j.neunet.2017.07.002, http://www.sciencedirect.com/science/article/pii/S0893608017301545.

[62] D. Yarotsky, *Optimal approximation of continuous functions by very deep ReLU networks*, in Proceedings of the 31st Conference On Learning Theory, S. Bubeck, V. Perchet, and P. Rigollet, eds., vol. 75 of Proceedings of Machine Learning Research, PMLR, 06–09 Jul 2018, pp. 639–649, http://proceedings.mlr.press/v75/yarotsky18a.html.