How Powerful are Shallow Neural Networks with Bandlimited Random Weights?

Ming Li [1] **Sho Sonoda** [2] **Feilong Cao** [3] **Yu Guang Wang** [4 5] **Jiye Liang** [6]

Abstract

We investigate the expressive power of depth-2 bandlimited random neural networks. A random net is a neural network where the hidden layer parameters are frozen with random assignment, and only the output layer parameters are trained by loss minimization. Using random weights for a hidden layer is an effective method to avoid non-convex optimization in standard gradient descent learning. It has also been adopted in recent deep learning theories. Despite the well-known fact that a neural network is a universal approximator, in this study, we mathematically show that when hidden parameters are distributed in a bounded domain, the network may not achieve zero approximation error. In particular, we derive a new nontrivial approximation error lower bound. The proof utilizes the technique of ridgelet analysis, a harmonic analysis method designed for neural networks. This method is inspired by fundamental principles in classical signal processing, specifically the idea that signals with limited bandwidth may not always be able to perfectly reconstruct the original signal. We corroborate our theoretical results with various simulation studies, and generally, two main take-home messages are offered: (i) Not any distribution for selecting random weights is feasible to build a universal approximator; (ii) A suitable assignment of random weights exists but to some degree is associated with the complexity of the target function.

1. Introduction

In recent years, there has been a growing interest in utilizing random methods for training neural networks as they have been shown to have the potential to significantly accelerate the training process, particularly for large-scale datasets and real-time processing requirements (Scardapane & Wang, 2017; Cao et al., 2018). In this study, we examine the capability of shallow random networks in a *bandlimited* context, where the hidden parameters are confined to a specific range. Past research has, in some cases, deliberately or unintentionally limited the distribution range of parameters. For instance, uniform distributions were used due to technical limitations, or normal random vectors with insufficiently small variance were employed due to the default configurations of the software. Analogous to the well-established principle in signal processing (or Fourier analysis) that bandlimited signals may not be able to accurately reproduce the original signal, bandlimited neural networks may not fully exhibit their function approximation capabilities. However, unlike classical Fourier analysis, the correlation between bandwidth and approximation error has yet to be clearly defined. One challenge in this area is that neural networks do not possess an orthonormal basis, but rather a frame. Through the use of ridgelet analysis, a Fourier-like analysis developed specifically for neural networks, we have derived a new lower bound for approximation error.

This study considers a shallow neural network $g_d(x) = \sum_{j=1}^{d} c_j \sigma(a_j \cdot x - b_j)$ of input $x \in \mathbb{R}^m$ with activation function σ and parameters $(a_j, b_j, c_j) \in \mathbb{R}^m \times \mathbb{R} \times \mathbb{R}$ for each $j \in [d] := \{1, \dots, d\}$, and the random training method with two steps:

> `Step I`: Randomly initialize hidden parameters (a_j, b_j) to a given data-independent probability distribution $Q(a, b)$, and freeze them; then
>
> `Step II`: Statistically determine output parameters c_j given a dataset $D_n = \{(x_i, y_i)\}_{i=1}^{n}$.

A variety of neural network architectures have been developed that utilize random training methods, including random vector functional-link (RVFL) networks (Igelnik & Pao, 1995), random feature expansions (Rahimi & Recht, 2008a;b; 2009), random weight networks (Saxe et al., 2011),

[1] Key Laboratory of Intelligent Education Technology and Application of Zhejiang Province, Zhejiang Normal University, Jinhua, China [2] Deep Learning Theory Team, RIKEN AIP, Tokyo, Japan [3] College of Sciences, China Jiliang University, Hangzhou, China [4] Institute of Natural Sciences, Shanghai Jiao Tong University, Shanghai, China [5] School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai, China [6] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan, China. Correspondence to: Ming Li <mingli@zjnu.edu.cn>, Sho Sonoda <sho.sonoda@riken.jp>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

stochastic configuration networks (Li et al., 2019; Li & Wang, 2021; Wang & Li, 2017a;b), graph convolutional networks with random weighths (Huang et al., 2023), and certain versions of over-parametrized networks (Belkin et al., 2019; Daniely, 2017; Ghorbani et al., 2019; Yehudai & Shamir, 2019). A kernel function defined by the inner product of feature maps: $k(x, x') = \int_{\mathbb{R}^m \times \mathbb{R}} \sigma(a \cdot x - b)\sigma(a \cdot x' - b) \mathrm{d}Q(a, b)$ is a special case of random training methods because we can regard this as a sum of infinitely many random samples $(a, b) \sim Q$ (Bach, 2017; Cho & Saul, 2009; Suzuki, 2018). On the other hand, Bayesian neural networks (Neal, 1996) are *not* strictly based on the random training method in consideration since the distribution Q is a "posterior", which contains the information of the dataset D_n; nor is the lazy learning (Jacot et al., 2018) since its hidden parameters are not strictly frozen.

The random training method has the remarkable trick of "convexification". It frees us from inevitable non-convexity in the standard gradient descent training. The non-convexity is caused by the hidden parameters (a_j, b_j) (and not by the-output parameters c_j). In the random training setting, we do not optimize the parameters in Step I, but only the output parameters c_j in Step II. This "randomization" trick is beneficial both for theory and applications, and has recently been adopted not only in practical algorithms but also in the theoretical study of deep learning (Belkin et al., 2019; Jacot et al., 2018; Louart et al., 2018; Malach et al., 2020; Pennington & Worah, 2017). While the potential of random neural networks has been explored, the expressivity of these networks has not been extensively studied. In this study, we aim to shed light on this topic by highlighting a limitation in expressivity of random nets. Our main contribution is the introduction of a new *approximation lower bound* for bandlimited shallow neural networks.

Main Theorem (simplified). *Let $\Omega \subset \mathbb{R}^m$ be a bounded open set with smooth boundary, and put $K := \overline{\Omega}$ be its closure. Suppose $f : \Omega \to \mathbb{R}$ be an L^2-Sobolev function on Ω with order at least $s \in (1/2, \infty]$. Let $\lambda > 0$, and put $V := \{(a, b) \in \mathbb{R}^m \times \mathbb{R} \mid |a| \le \lambda, |b| \le \lambda\}$. Suppose $\sigma : \mathbb{R} \to \mathbb{R}$ is bounded, self-admissible, and the constant $C_{\sigma,P} := \sup_{(a,b)\in V} \|\sigma_{a,b}\|_{L^2(K)}$ exists finite. Consider approximating f with a bandlimited shallow neural network $g_d(x) = \sum_{j=1}^d c_j \sigma(a_j \cdot x - b_j)$. Then, the approximation error is lower bounded as*

$$\|f - g_d\|^2_{L^2(K)} \ge \|S^*[f]\|^2_{L^2(V^c)} \ge \|f\|^2_{L^2(K)} - C_b^2 \begin{cases} \|f\|^2_{L^1(K)} \|\sigma\|^2_{L^\infty(\mathbb{R})} \lambda^m, & \lambda \in (0, \vartheta) \\ \|f\|^2_{H^s(\Omega)} C^2_{\sigma,s} \left(\frac{-1}{2s} \lambda^{-2s} + \frac{2s+m}{2sm} \vartheta^{-2s}\right), & \lambda \in [\vartheta, \infty) \end{cases}.$$

Here, S^* is the adjoint of integral representation operator S; and the constants $C_b, C_{\sigma,s}$ and threshold $\vartheta > 0$ depend on norms of f, dimension m and smoothness s. (See Figure 1 for the outline, and Section 3 for more detailed statement.)

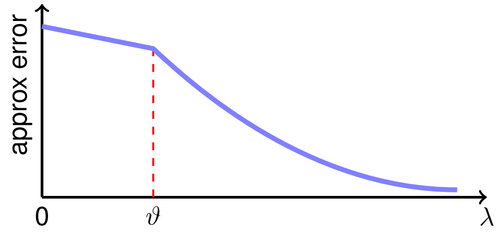


Figure 1. Outline of the approximation lower bound.

Our results, as depicted in Figure 1, demonstrate that the lower bound: (i) holds true regardless of whether the hidden parameters are random or deterministic, (ii) is always nonnegative, (iii) converges to 0 as bandwidth λ tends to ∞, which agrees with the universality of the network, (iv) is continuous at $\lambda = \vartheta$, and (v) depends on the smoothness s of target function f when $\lambda \ge \vartheta$. This implies that the lower bound is more significant when the target function is smoother. Thus, if the domain of hidden parameters is *bandlimited*, the approximation error may not reach 0. To better illustrate this interesting phenomenon, we provide quantitative demonstration based on two simple examples in Appendix D.3.

Additional remarks to avoid potential confusions. In the **Main Theorem**, *randomness is not essential/required but bandlimiting is.* Nevertheless, we focus on random nets because we can find a lot of usecases both in theorertical and practical aspects. As long as the hidden unit number is finite, random nets are always bandlimited. We may list two more examples: (1) Random nets with fully-supported proposal distribution Q (such as Gaussian), because hidden unit parameters are in a bounded domain with high probability. (2) NNs trained by gradient descent in the lazy learning regime, because the lazy assumption (that final parameters are close to the initial parameters) means that the final parameters are distributed in a bounded domain (with high probability).

Notation. For any complex number z, we denote by $\overline{z}$ the complex conjugate of z. For any subset $A \subset X$ of a set X, $A^c (= X \setminus A)$ denotes the complement of A. For an activation function σ, we write $\sigma_{a,b}(x) := \sigma(a \cdot x - b)$. For any vector $x \in \mathbb{R}^d$, we denote $\langle x \rangle := (1 + |x|^2)^{1/2}$, where $|x|$ is the Euclidean norm of x. $\mathcal{S}(\mathbb{R})$ denotes the space of rapidly decreasing smooth functions, or the Schwartz test functions, on $\mathbb{R}$; and $\mathcal{S}'(\mathbb{R})$ denotes the space of tempered distributions, or the topological dual space of $\mathcal{S}(\mathbb{R})$. $H^s(\Omega)$ denotes the L^2-Sobolev space on open set Ω of order $s (\in \mathbb{R})$. In order to avoid confusion, we use $\rho^\sharp(\omega) := \int_{\mathbb{R}} \rho(t) \exp(-it\omega) \mathrm{d}t$ for 1-dimensional Fourier transform of $\rho \in L^2(\mathbb{R})$, and $\widehat{f}(\xi) := \int_{\mathbb{R}^m} f(x) \exp(-i\xi \cdot x) \mathrm{d}x$ for m-dimensional Fourier transform of $f \in L^2(\mathbb{R}^m)$. By the terms 'random neural networks', 'random nets', or 'neural nets with random weights' we mean the same thing.

2. Integral Representation and Ridgelet Transform

In this section, we introduce a few basics of the integral representation theory and ridgelet analysis, then provide several important (known) propositions that will be used in proving our main results. In Appendix B, we further supplemented the backgrounds and detailed (but quick) overview.

2.1. Integral Representation of Neural Nets

The integral representation is a handy tool for the analysis of neural networks with a variable number of hidden units.

Let V be a Borel subset in $\mathbb{R}^m \times \mathbb{R}$, and $\mathcal{M}(V)$ be the space of all signed Radon measures on V. We set V to be a space of hidden parameters (a, b), and call an element $\mu \in \mathcal{M}(V)$ a *parameter distribution*.

Let $\sigma : \mathbb{R} \to \mathbb{R}$ be an *activation function*. We always assume that the activation function σ is associated with a function ρ that satisfies the admissibility condition, which is a sufficient condition for the neural network to have the universal approximation property (see § 2.2 and Proposition 1).

Definition 1. The *integral representation* of a neural network is defined as an integral transform of the parameter distribution $\mu \in \mathcal{M}$:

$$S[\mu](x) := \int_{\mathbb{R}^m \times \mathbb{R}} \sigma(a \cdot x - b)\mathrm{d}\mu(a, b), \quad x \in \mathbb{R}^m. \quad (1)$$

We have two motivations to employ the integral representation introduced above. First, it provides a unified expression for a variable number d of parameters including an infinite number of parameters. As the integral suggests, it is *formally* an infinite version of the ordinary finite neural network. Namely, whereas the finite net $g_d = \sum_{j=1}^d c_j \sigma(a_j \cdot x - b_j)$ is a weighted sum of a finite number of hidden units $\sigma(a_j \cdot x - b_j)$ with weights c_j and indices $j \in [d]$, the infinite net $S[\mu]$ is a weighted integral of an infinite number of hidden units $\sigma(a \cdot x - b)$ with weight function $\mu(a, b)$ and "indices" $(a, b) \in V$. Nevertheless, we can also *express a finite net* as $g_d = S[\mu_d]$ by letting $\mu_d = \sum_{j=1}^d c_j \delta_{(a_j, b_j)}$ with Dirac measures $\delta_{(a_j, b_j)}$, because we assume that a parameter distribution μ is a Radon measure, which includes both continuous densities and singular masses. In other words, the integral representation is *not* a counterpart of the finite models, but it is an extension of the finite models. Second, the map S is linear in μ. Since the non-linear parameters (a, b) are "integrated out" in the integral representation (as "marginalized out" in the Bayesian literature), we are now free from the non-linearity of neural networks.

2.2. Admissibility Condition

Definition 2. Given an activation function $\sigma : \mathbb{R} \to \mathbb{C}$, we say that a function $\rho : \mathbb{R} \to \mathbb{C}$ is *admissible* when it satisfies the *admissibility condition*

$$(2\pi)^{m-1} \int_{\mathbb{R}} \sigma^\sharp(\omega)\overline{\rho^\sharp(\omega)}|\omega|^{-m}\mathrm{d}\omega = 1. \quad (2)$$

This condition seems technical, but is typical in the context of wavelet transforms (see e.g., Mallat (2009)). It simply requires the $|\omega|^{-m}$-weighted inner product of $\sigma^\sharp$ and $\rho^\sharp$ to be finite (not zero nor infinite). Therefore, this is not a strong condition and we can find, in general, an infinite number of different ρ's for a given σ. For example, if σ is a gaussian, then its Fourier transform $\sigma^\sharp$ is again another gaussian, and we can find a "family of" particular solutions: $\rho^\sharp(\omega) = C|\omega|^m \phi^\sharp(\omega)$ with an arbitrary Schwartz function $\phi \in \mathcal{S}(\mathbb{R})$ (as long as the integral is finite and not zero) and an appropriate normalizing constant $C > 0$. We refer to (Sonoda & Murata, 2017, § 6.2) for more examples. Finally, when $\rho = \sigma$, we say that ρ (or σ) is *admissible with itself*, or *self-admissible*.

2.3. Ridgelet Transform

The ridgelet transform R is, in a nutshell, a right inverse operator to the integral representation operator S. Given a function $f \in L^2(\mathbb{R}^m)$, consider finding an unknown parameter distribution $\mu \in \mathcal{M}$ that satisfies the integral equation $S[\mu] = f$. As we would describe later, the solution to this integral equation is not unique, and the ridgelet transform provides a particular solution to this equation.

Definition 3. For every $f \in L^p(\mathbb{R}^m) (p = 1, 2)$, the *ridgelet transform* of f with respect to $\rho \in \mathcal{S}(\mathbb{R})$ is given by

$$R[f](a, b) := \int_{\mathbb{R}^m} f(x)\overline{\rho(a \cdot x - b)}\mathrm{d}x, \quad (a, b) \in \mathbb{R}^m \times \mathbb{R}.$$

We provide two important propositions as a basic preparation for the main theoretical analysis performed in Section 3. Their proofs are reported in Appendix B.

Proposition 1 (Reconstruction formula)**.** *Let $f \in L^p(\mathbb{R}^m) (p = 1, 2)$. Provided that $\rho \in \mathcal{S}(\mathbb{R})$ is admissible with an activation function $\sigma \in \mathcal{S}'(\mathbb{R})$, then we have*

$$\begin{aligned} S[R[f]](x) &= \int_{\mathbb{R}^m \times \mathbb{R}} R[f](a, b)\sigma(a \cdot x - b)\mathrm{d}a\mathrm{d}b \\ &= f(x), \quad x \in \mathbb{R}^m. \end{aligned}$$

We have two interpretations for Proposition 1. First, recall that $S[\mu]$ represents a neural network. Then, the reconstruction formula implies the *universal approximation property*, because a neural network $S[\mu]$ can express any function $f \in L^p(\mathbb{R}^m)(p = 1, 2)$ by letting $\mu = R[f]$. Second, recall the Fourier inversion formula: $F^{-1}[\widehat{f}](x) = (2\pi)^{-m} \int_{\mathbb{R}^m} \widehat{f}(\xi) \exp(i\xi \cdot x) \mathrm{d}\xi = f(x)$. Then, we can find a clear correspondence of S with F^{-1}, $R[f]$ with $\widehat{f}$, and $\sigma(a \cdot x - b)$ with $\exp(i\xi \cdot x)$. However, we should also remark the difference that by the non-uniqueness of admissible functions ρ, the ridgelet transform $R[f]$ is not unique either. This means that R is *not* the strict inverse operator to S, but only a right inverse operator to S.

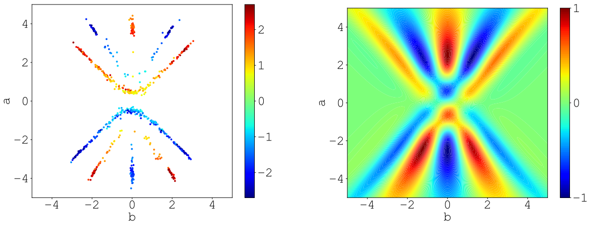
Proposition 2 (Plancherel theorem)**.** *Let $f \in L^2(\mathbb{R}^m)$. Provided that σ is self-admissible, one has $\|R[f]\|_{L^2(\mathbb{R}^m \times \mathbb{R})} = \|f\|_{L^2(\mathbb{R}^m)}$.*

The Plancherel theorem plays a key role in our main results. As to be displayed in Figure 2, a ridgelet spectrum $R[f]$ has a long tail. If the ridgelet spectrum $R[f]$ is truncated, then the Plancherel theorem implies that we cannot reconstruct f without loss.

2.4. Proof Idea Behind Main Theorem

To enhance the readers' understanding of our main theorem, we present a visual example of *how real parameters are distributed*. Without this visualization, some readers may imagine/assume that neural network parameters are distributed randomly, with relatively simple structures such as uniform distribution and normal distribution. On the contrary, *the parameter distribution has the structure of ridgelet transform.* With this illustration, we can intuitively/visually understand how/why the expressive power is lost when the parameter distribution is truncated to a bounded domain.

Figure 2, produced by Sonoda et al. (2018), visualizes parameter distributions in two ways.



(a) GD trained parameters

(b) Ridgelet spectrum

Figure 2. Example of parameter distributions

Both figures are obtained from the common dataset $D_n = \{(x_i, y_i)\}_{i=1}^n$ that is generated by the function $y = f(x) = \sin 2\pi x$. Figure 2(a) shows the distribution of the parameters (a_j, b_j, c_j), which are obtained from many neural networks trained on the common dataset D_n by gradient descent (GD); and Figure 2(b) shows the ridgelet spectrum $R[f](a, b)$ approximated by numerical integration evaluated at each point (a, b).

Despite the fact that two figures are obtained from different procedures, gradient descent and numerical integration, they have an apparent intriguing resemblance. The shapes of the distributions are 10-point star shaped. In other words, the trained parameters (a_j, b_j, c_j) concentrate on high intensity areas in the ridgelet spectrum. This phenomenon that GD converges to the ridgelet spectrum is initially reported by Sonoda et al. (2018), and recently given a mathematical justification by Sonoda et al. (2021a).

Based on the visualized results, one can naturally *conjecture* that if the parameter space is bandlimited, that is, the spectrum is truncated to a compact domain such as $|a| \leq \lambda$ and $|b| \leq \kappa$, then the neural network loses the universal approximation property. In other words, there exists a class of functions that a bandlimited network cannot reconstruct. Overall, that is the primary idea behind our main theorem, i.e., we quantify and prove this conjecture by carefully estimating the tail of the ridgelet spectrum.

3. Main Results

For theoretical analysis, we reformulate the random training method at the beginning of the introduction as follows: Let Ω be a bounded open set with smooth boundary, put $K := \overline{\Omega}$, and let V be a Borel set in $\mathbb{R}^m \times \mathbb{R}$.

> `Step I′`: Let $\{(a_j, b_j)\}_{j=1}^d$ be arbitrary d points in V, and let $\mathcal{M}(d) := \{\sum_{j=1}^d c_j \delta_{(a_j, b_j)} \mid c_j \in \mathbb{R}\}$,
>
> `Step II′`: Let $\mu_d^\circ := \arg\min_{\mu \in \mathcal{M}(d)} \|f - S[\mu]\|_{L^2(K)}^2$, and let $g_d^\circ := S[\mu_d^\circ]$.

Here, the generation process of (a_j, b_j) need not be random as long as these are inside of V.

The main goal of this section is to lower bound the approximation error $\|f - g_d^\circ\|_{L^2(K)}$. This is also a lower bound on $\|f - g_d\|_{L^2(K)}$, where g_d is provided by Step II in Section 1, since $\|f - g_d^\circ\|_{L^2(K)} \leq \|f - g_d\|_{L^2(K)}$ by construction. Unlike the Fourier or Taylor series expansions, the rate of approximation lower bound for a finite d is unknown, and it is known as a (complicated) open question (see Kainen et al. (2013) for more details). To circumvent this difficulty, we only estimate the approximation error achieved by infinite minimizers, $\mu \in \mathcal{M}(V)$, which exists as a consequence of the extreme value theorem, and lower bounds the approximation error achieved by its finite minimizers as follows:

$$\begin{aligned} \|f - g_d^\circ\|_{L^2(K)}^2 &= \min_{\mu \in \mathcal{M}(d)} \|f - S[\mu]\|_{L^2(K)}^2 \\ &\geq \inf_{\mu \in \mathcal{M}(V)} \|f - S[\mu]\|_{L^2(K)}^2, \end{aligned} \quad (3)$$

where the inequality above is an immediate consequence of the inclusion $\mathcal{M}(d) \subset \mathcal{M}(V)$. Namely, just contrary to the case of estimating upper bounds, the lower bound for more expressive (=infinite) networks is automatically valid for less expressive (=finite) networks.

In Theorem 1, we show that the infinite minimum is lower bounded by the tail part of the ridgelet spectrum.

Theorem 1. *Let $f \in L^2(K)$ be a square-integrable function that is supported in the compact domain K. Assume that σ is self-admissible, and that the constant $C_{\sigma,P} := \sup_{(a,b)\in V} \|\sigma_{a,b}\|_{L^2(K)}$ exists finite. Then, the approximation error is lower bounded as follows:*

$$\inf_{\mu\in\mathcal{M}(V)} \|f - S[\mu]\|^2_{L^2(K)}$$
$$\geq \|S^*[f]\|^2_{L^2(V^c)} = \int_{V^c} |S^*[f](a,b)|^2 \mathrm{d}a\mathrm{d}b, \qquad (4)$$

where S^* is the adjoint operator of S.

Namely, if the tail part $S^*[f]|_{V^c}$, or the ridgelet spectrum outside the parameter domain V, does not vanish, then the tail bound $\|S^*[f]\|^2_{L^2(V^c)}$ inevitably lower bounds the approximation error $\|f - g_d^\circ\|_{L^2(K)}$.

In order to quantify (or estimate from below) the tail bound $\|S^*[f]\|_{L^2(V^c)}$, we exploit the property $S^*[f] = R[f]$ (valid in case of a self-admissible function σ, see Appendix B), and rewrite the right-hand side of (4) as

$$\|S^*[f]\|^2_{L^2(V^c)} = \|R[f]\|^2_{L^2(V^c)} = \|f\|^2_{L^2(K)} - \|R[f]\|^2_{L^2(V)}.$$

Then, the estimation problem of $\|S^*[f]\|_{L^2(V^c)}$ from *below* is now turned to the estimation problem of $\|R[f]\|_{L^2(V)}$ from *above*. Thus, we can estimate the tail bound through the decay property of ridgelet spectrum, which is given by the following theorem.

Theorem 2. *Let $f \in H^s(\Omega)$ be an L^2-Sobolev function on Ω with order $s \in (1/2, \infty]$. Assume that $\rho \in L^\infty(\mathbb{R})$ be self-admissible. For $(r,u,b) \in \mathbb{R}_+ \times \mathbb{S}^{m-1} \times \mathbb{R}$, put*

$$\phi_a(ru) := \min\left\{ \|f\|_{L^1(K)} \|\rho\|_{L^\infty(\mathbb{R})}, C_{\rho,s} \Phi_s[f](u) r^{\frac{-2s-m}{2}} \right\},$$
$$\phi_b(ru,b) := |R[f;\rho](ru,b)|/\phi_a(ru) \quad (\leq 1),$$

where $C^2_{\rho,s} := \frac{2}{(2\pi)^2} \int_{\mathbb{R}} \langle\omega\rangle^{-2s+1} |\rho^\sharp(\omega)|^2 |\omega|^{-m} \mathrm{d}\omega$, which always exists finite; and $\Phi_s[f](u)^2 := \int_0^\infty \langle\omega\rangle^{2s} |\widehat{f}(\omega u)|^2 \omega^{m-1} \mathrm{d}\omega$, which satisfies $\int_{\mathbb{S}^{m-1}} \Phi_s[f](u)^2 \mathrm{d}u = \|f\|^2_{H^s}$. Then, the ridgelet spectrum is upper bounded by

$$|R[f;\rho](ru,b)| \leq \phi_a(ru), \quad (r,u,b) \in \mathbb{R}_+ \times \mathbb{S}^{m-1} \times \mathbb{R}.$$

Furthermore, when V is given by a product $V_a \times V_b$ with some $V_a \subset \mathbb{R}^m$ and $V_b \subset \mathbb{R}$, we can decompose the integral:

$$\|R[f;\rho]\|_{L^2(V_a \times V_b)} = \|\phi_a\|_{L^2(V_a)} \|\phi_b\|_{L^2(V_b)}.$$

Finally, by integrating the dominating functions ϕ_a and ϕ_b, we obtain an estimate of the tail bound $\|R[f]\|_{L^2(V^c)}$, as follows.

Theorem 3 (Main Theorem)**.** *Let $f \in H^s(\Omega)$ with $s \in (1/2, \infty]$. Let $\lambda > 0$, $V_a := \{a \in \mathbb{R}^m \mid |a| \leq \lambda\}$ be a ball, $V_b \subset \mathbb{R}$ be an arbitrary Borel set, and put $V = V_a \times V_b$. Assume that $\sigma \in L^\infty(\mathbb{R})$ is self-admissible, and that the constant $C_{\sigma,P} := \sup_{(a,b)\in V_a\times V_b} \|\sigma_{a,b}\|_{L^2(K)}$ exists finite. Put $\vartheta := (mV_m \|f\|^2_{L^1(K)} \|\rho\|^2_{L^\infty(\mathbb{R})} / \|f\|^2_{H^s(\Omega)} C^2_{\rho,s})^{-1/(2s+m)}$, where $V_m := \pi^{m/2}/\Gamma(m/2+1)$ is the volume of the m-unit ball. Then, for a bandlimited network $g_d = \sum_{j=1}^d c_j \sigma(a_j \cdot x - b_j)$ obtained by Steps I' and II', we have the following approximation lower bounds:*

$$\|f - g_d\|^2_{L^2(K)} \geq \inf_{\mu\in\mathcal{M}(V)} \|f - S[\mu]\|^2_{L^2(K)}$$
$$\geq \|S^*[f]\|^2_{L^2(V^c)} = \|f\|^2_{L^2(K)} - \|\phi_b\|^2_{L^2(V_b)} \cdot$$
$$\begin{cases} \|f\|^2_{L^1(K)} \|\sigma\|_{L^\infty(\mathbb{R})} \lambda^m, & \lambda \in (0,\vartheta), \\ \|f\|^2_{H^s(\Omega)} C^2_{\sigma,s} \left(\frac{-\lambda^{-2s}}{2s} + \frac{2s+m}{2sm} \vartheta^{-2s} \right), & \lambda \in [\vartheta, \infty). \end{cases}$$

Here, the final bound is continuous at $\lambda = \vartheta$, always nonnegative, and tends to 0 as $\lambda \to \infty$.

We provide the proofs of all the theoretical results above in Appendix C.

3.1. Technical Remarks

(Un)necessity of Randomness. Even though our subject is random nets, we *do not need any randomness in the main theorem* because the key inequality (3) holds for any realization of $\mu \in \mathcal{M}(V)$ (besides that the function f is fixed). According to Steps I' and II', the LHS of (3) is a random variable. However, the RHS is not a random variable but a constant because it is by definition smaller than any loss-value $J(\mu) := \|f - S[\mu]\|^2_{L^2(K)}$ for $\mu \in \mathcal{M}(V)$. (4) (= RHS of (3)) indicates that the lower bound on $\underline{J} := \inf_{\mu\in\mathcal{M}(V)} J(\mu)$ is strictly positive when the ridgelet spectrum $R[f]$ is supported on a set containing the parameter domain V. Thus, if the proposal distribution $Q(a,b)$ (in Step I) is supported on a compact set V and $R[f]$ has support containing V, then inevitably $\underline{J} > 0$. We may consider extensions to a fully supported distribution such as the normal distribution N. For this case, in contrast, to extend our main result, *we need some high probability condition* that initial parameters $\{(a_j, b_j)\}_{j=1}^d$ concentrate on a certain compact domain V.

Extension to Unbounded Activation Functions such as ReLU. It would be possible, but not immediate. The Plancherel formula (Proposition 2) is a key step to obtain the lower bound in Theorem 1, and the self-admissible assumption in Proposition 2 is the main cause of the bounded assumption on activation function. Recently, Sonoda et al. (2021b) have extended the Plancherel formula for unbounded activation functions. Thus, it is technically possible and left for our future work to derive the lower bound for unbounded activation functions by similar arguments.

Extension to Deep. The ridgelet theory is essentially based on the linearity of parameter distribution μ in the integral representation $S[\mu]$. But this linearity is specific to the single-hidden-layer structure. Namely, in a DNN such as $S[\mu_2] \circ S[\mu_1]$, the μ_1 (inside the σ of $S[\mu_2]$) is no more linear. Technically, we need a deep ridgelet theory, but there are no such things yet. We note that other theories based on the linearity of shallow networks, such as the mean field theory, also face to the same difficulty.

Verification of Tightness. In fact, the obtained lower bound is not tight, simply because NNs with $M(V)$ (infinitely many hidden units) are more expressive than NNs with $M(d)$ (at most d hidden units). (On the other hand, for the infinitely-many-hidden-units case $M(V)$, the Pythagorean relation (7) is tight.) Nonetheless, we consider this relaxed bound meaningful because we can interpret the bound as: If the band is limited, then even if we use infinite units, the approximation power can be limited.

Estimation of Upper Bound. We consider it an out-of-scope because (1) estimating the approximation error with respect to *finite* unit number d *with* bandlimiting assumption is another challenging problem, and (2) our focus is to present a non-trivial lower bound (since sometimes random nets are misunderstood to be always universal). In fact, before this study, there was no lower bound for a bandlimited network, even though it sounds reasonable when we consider the Fourier series expansion. And the difficulty why it has not been shown is the existence of null components, which Fourier series expansion does not hold. For the case of *finite* hidden units *without* bandlimiting assumption, two types of upper bounds—the Jackson bound $O(d^{-s/m})$ and the Maurey-Jones-Barron (MJB) bound $O(1/\sqrt{d})$ were obtained by multiple authors in the 1990s. However, these upper bounds are in general not sharp for bandlimiting cases.

4. Related Work and Further Remarks

For a whole picture, we should recall the pioneering work by Barron, Theorem 6 in (Barron, 1993), which is a lower bound on the best approximation error for linear combinations of any *fixed* basis functions:

$$\inf_{(a_j,b_j)} \sup_{C_f \le C} \inf_{c_j} \|f - g_d\|_{L^2([0,1]^m)} \ge \frac{\kappa C}{m d^{1/m}},$$

where C_f is a certain complexity of function f, κ is a universal constant not smaller than $1/(8\pi e^{\pi-1})$ (further refinements/improvements can also be found in (Gnecco, 2012; Kůrková & Sanguineti, 2002)), m denotes the input dimension, d stands for the number of hidden neurons. Barron's theoretical results, related to the so-called *Kolmogorov width*, indicate that "*fixed basis function expansions must have a worst-case performance that is much worse than that which is proven to be achievable by certain adaptable basis function methods (such as neural nets)*." We note that *neural nets with random frozen weights* is a special case of *fixed basis function expansion*. However, for fixed C and a given approximation error tolerance, the estimate $\kappa C m^{-1} d^{-1/m}$ goes to 0 as either m or d tends to infinity; in this case, the lower bound is of impractical use to show the smaller effectiveness of fixed basis function approximation. Similarly, Yarotsky (2017) considered the problem that a deep ReLU net (not random but in which all the parameters are adaptable, without any norm constraints on the weights) approximates an L^∞-Sobolev function $f \in W^{s,\infty}([0,1]^m)$. Based on covering number arguments, he proved (in Theorem 5) that if a ReLU net ϵ-approximates f in a unit ball, i.e. $\|f\|_{W^{s,\infty}([0,1]^m)} \le 1$, then the ReLU net must have at least $d_0 = c\varepsilon^{-m/9s}$ units:

$$\sup_{\|f\|_{W^{s,\infty}([0,1]^m)} \le 1} \inf_{params.} \|f - g_d\|_{L^\infty([0,1]^m)} \ge \frac{C}{(md)^{9s/m}}.$$

However, this again goes to 0 as either m or d tends to infinity. The difference lies in the assumptions on the approximator g_d and approximated function f. The Kolmogorov width considers the setting where g_d is not limited and f is the worst one and thus the bound is uniform, while our result considers the setting where g_d is bandlimited and f is an arbitrary given one and thus the bound is pointwise.

In the context of modern deep learning theory, Yehudai & Shamir (2019) and Ghorbani et al. (2019) proved (under very limited settings) that the expressive power of random nets is low, while Malach et al. (2020) proved a stronger lottery ticket hypothesis, which essentially claims that the expressive power is exceptionally high. These seemingly contradictory claims are, of course, consistent. Yehudai & Shamir (2019) considered the problem that a finite-dimensional random net (FRN) approximates a single ReLU neuron and provided an approximation lower bound w.h.p. for a finite number of parameters d to conclude low expressive power. Ghorbani et al. (2019) considered the problem that an FRN approximates a quadratic function and showed that the asymptotic approximation error does not tend to zero (Theorem 1). Namely, these two studies focused on specific examples that FRNs *cannot* easily approximate. On the other hand, Malach et al. (2020) considered the so-called student-teacher problem in which a student FRN approximates teacher FRN, and proved that if both the student and the teacher share a common norm constraint, then the student can ϵ-approximate the teacher w.h.p., which does not contradict the previous two (and our) claims because this study focused on specific examples that FRNs *can* easily approximate. Hsu et al. (2021) studied the approximation power of two-layer networks of random ReLUs, where both upper and lower-bounds for Lipschitz functions with explicit asymptotics were provided. However, the role of the hyper-parameter λ, determining the selection range of the random weights (and biases), is not considered in their main theorems, in contrast to our Theorem 3. Compared to these results, we consider the problem in which a potentially infinite-dimensional random net approximates an L^2-Sobolev function $f \in H^s(\Omega)$ and provide an approximation error lower bound. Thus, our results cover a wider range of functions than previous studies.

5. Numerical Experiments

In this section, we conduct some simulation studies to verify our theoretical results. Two toy examples for 1D function regression are used in our experiments. Consistent with our theoretical analysis, the numerical simulations aim at showing how λ, which is used for the random assignment of input weights (and biases), would affect the expressive power of the random net. For this purpose, we present an intuitive illustration of the infeasibility of individual trivial settings of λ. Then, we would discuss statistically the potential relationship between λ and the critical parameter that can determine the complexity of the target function. We utilize the following 1D target function in the following Simulation 1 and Simulation 2.

$$f(x;\sigma)=0.2\exp\left(-\frac{(x-0.4)^2}{\sigma^2}\right)+0.5\exp\left(-\frac{(x-0.6)^2}{\sigma^2}\right),$$

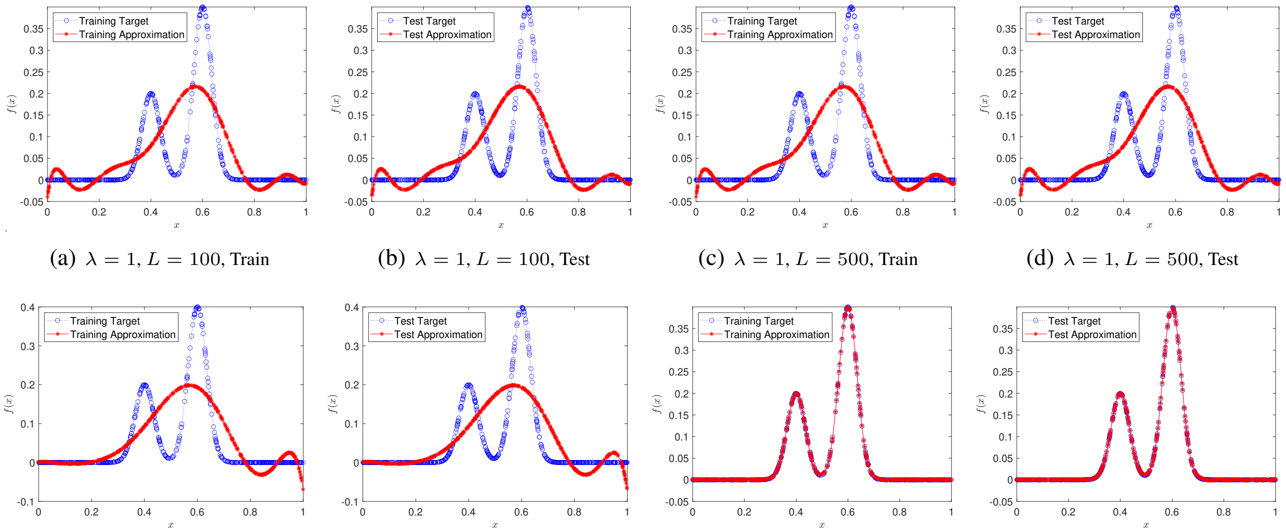
where $x \in [0,1]$, $\sigma > 0$ is a scalar index that can determine the complexity of f, as mentioned in our theoretical analysis. In Simulations 1 and 2, we use the sigmoid activation function.

Simulation 1. We set $\sigma = 0.05$ and sample 1000 instances $\{x_i, f(x_i)\}_{i=1}^{1000}$ based on equally spaced points on [0,1], then randomly and uniformly select 500 training sample and 500 test samples. We test the performance of two random networks with $\lambda = 1$ and $\lambda = 20$. For each case, we train the network with a different number of hidden nodes L, which helps with excluding the influence of L to our analysis. In Figure 3, we show the training and test approximation results for four different random networks, including (a) and (b) for the network built with $\lambda = 1, L = 100$, (c) and (d) for the network built with $\lambda = 1, L = 500$, (e) and (f) for the network built with $\lambda = 1, L = 10000$, (g) and (h) for the network built with $\lambda = 20, L = 200$, respectively. We observe that the random network with $\lambda = 1$ cannot achieve a good approximation accuracy for this simple function approximation problem, even when the number of hidden nodes is sufficiently large. In contrast, the network with $\lambda = 20$ and trained with $L = 200$ demonstrates excellent learning and generalization performance. Other larger values of λ, such as $\lambda = 50, 100, 150, 200$ as we tested, have the same excellent performance on this regression task. This implies that the choice of λ has a strong impact on the random network's expressive power, which is consistent with our theoretical results.

Simulation 2. Following the intuitive investigation of the role of λ in the expressive power of random networks in **Simulation 1**, in this part, we present more statistical results for function approximation with various pairs of (λ, σ) so that we can summarize a general pattern empirically. Specifically, we create different forms of target function $f(x;\sigma)$ by choosing σ as one element of the set $\{0.01, 0.05, 0.1, 0.5\}$, and for each regression task we build random nets with λ taken as an element from the set $\{0.1, 0.5, 1, 5, 10, 50, 100, 200\}$, and choose a sufficiently large L (here, $L = 10000$ in each case) so that we can observe the trend as $L \to +\infty$. In a similar way as in Simulation 1, we sample 1000 instances $\{x_i, f(x_i)\}_{i=1}^{1000}$ which are equally spaced points on [0,1], then randomly and uniformly select 500 training samples and 500 test samples. For each pair (λ, σ), we run independently 50 trials and calculate the relative training error $E_k := \|\vec{f} - \vec{y}\|_2 / \|\vec{f}\|_2$ for each trial, where $k = 1, 2, \ldots, 50$, $\vec{f} = (f(x_1), f(x_2), \ldots, f(x_{500}))$ represents the vector of training targets, $\vec{y} = (y_1, y_2, \ldots, y_{500})$ stands for the output vector of the random network. As a matter of fact, as already shown by Figure 3, we only need to study the training performance to see whether a given λ is suitable for approximating the target function produced by a given σ. Table 1 summarizes the averaged relative training error $E := \sum_{k=1}^{50} E_k/50$. Note that we do not provide their standard deviations here because, compared with the average values, standard deviation values may not affect the conclusion that we are aiming to verify, as we will detail later. Table 1 shows how the choice of (σ, λ) affects the approximation ability of random networks. From the colored cells of the table, which values are tiny (magnitude between e-8 and e-6), we can observe that, for a target function with a smaller σ value, we would need a larger λ for a random net to ensure a random network to achieve an accurate approximation of the target function. From the above simulations, we can see that the effectiveness of the approximation by

Table 1. Summary of mean relative training error for various choices of (λ, σ).

λ	Averaged Relative Training Error E			
	$\sigma = 0.01$	$\sigma = 0.05$	$\sigma = 0.1$	$\sigma = 0.5$
$\lambda = 0.1$	0.9504	0.6969	0.3149	0.0026
$\lambda = 0.5$	0.9299	0.6627	0.2179	1.0606e-04
$\lambda = 1$	0.9188	0.6546	0.2089	1.1781e-05
$\lambda = 5$	0.8574	0.1263	0.0016	5.8661e-08
$\lambda = 10$	0.5714	0.0064	5.5692e-08	4.5881e-08
$\lambda = 50$	0.0131	4.4905e-08	4.6897e-08	4.5834e-08
$\lambda = 100$	1.9055e-06	7.5046e-08	7.2133e-08	6.8683e-08
$\lambda = 200$	1.1171e-07	1.3937e-07	1.0784e-07	1.1284e-07



(a) $\lambda = 1$, $L = 100$, Train (b) $\lambda = 1$, $L = 100$, Test (c) $\lambda = 1$, $L = 500$, Train (d) $\lambda = 1$, $L = 500$, Test

(e) $\lambda = 1$, $L = 10000$, Train (f) $\lambda = 1$, $L = 10000$, Test (g) $\lambda = 20$, $L = 200$, Train (h) $\lambda = 20$, $L = 200$, Test

Figure 3. Performance of random nets with $\lambda = 1$ and $\lambda = 10$ in training and test. (a-b) $\lambda = 1, L = 100$. (c-d) $\lambda = 1, L = 500$. (e-f) $\lambda = 1, L = 10000$. (g-h) $\lambda = 20, L = 200$.

random networks is constrained by both the network parameter distribution and the class of target functions. For a given learning task, there exists an appropriate range/distribution $\mathcal{D}^*$ (not unique), but **NOT ANY** range/distribution, such that a neural network with random weights assigned from $\mathcal{D}^*$ can be a universal approximator (if the number of hidden nodes is sufficiently large). Second, the $\mathcal{D}^*$ (for example, $[-\lambda^*, \lambda^*]$) is highly dependent upon the complexity of the target function. One needs an adequate amount of samples from the target function to provide some prior knowledge or empirical studies to discover $\mathcal{D}^*$.

Simulation 3. To further reveal the infeasibility of the trivial range [-1,1] for certain function approximation problems, we conduct similar simulations on a new target function $g(x)$, denoted as

$$g(x) = 0.5\cos(22\pi x^2) + 0.5x^2,\ x \in [0, 1].$$

Mathematically, $g(x)$ is composed of two parts: $0.5\cos(22\pi x^2)$ and $0.5x^2$, which represent two completely different 'modes' at distinct 'frequencies', as shown in Figure 4.

We carry on the same sampling as in Simulations 1 and 2 to generate 500 training and test points on [0,1]. Here, we only consider the training performance of random nets with various choices of λ. We report the results of the comparison for $\lambda = 1$ and $\lambda = 100$ in Figure 5. We observe that the random net with $\lambda = 1$ is not a universal approximator, although the number of hidden nodes is sufficiently large ($L = 10,000$). The network with $\lambda = 1$ can only fit the 'mean' curve of the original signal and fails to approximate the high-frequency 'mode' $0.5\cos(22\pi x^2)$. On the other hand, for the second 'mode' $0.5x^2$, the random net with $\lambda = 1$ has great approximation performance.

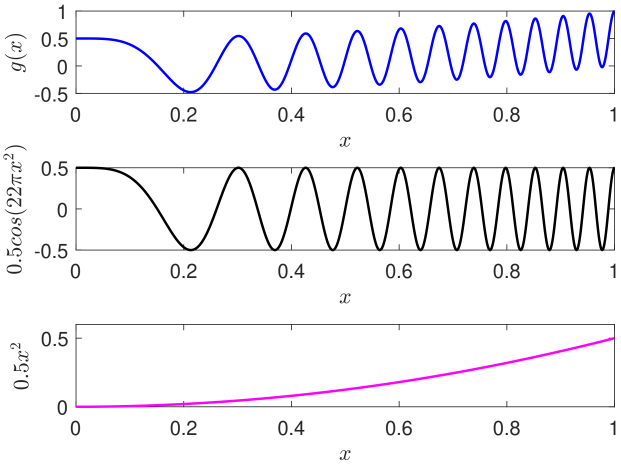
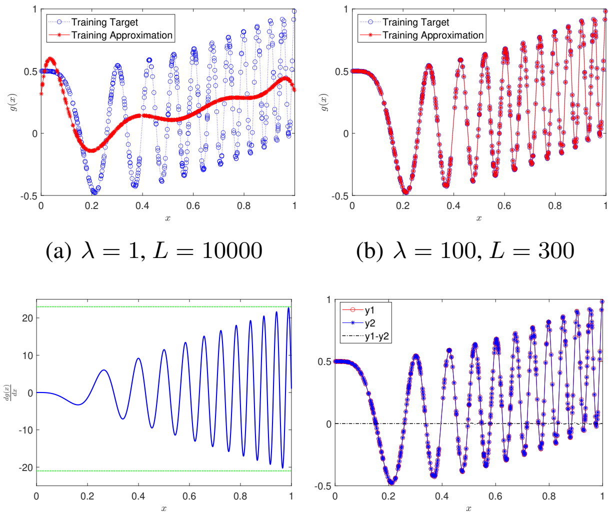


Figure 4. Visualization for the target function $g(x)$ (up), its modes $0.5\cos(22\pi x^2)$ (middle) and $0.5x^2$ (bottom), respectively.

As we observe the derivative $|g'(x)| \leq 25$ in Figure 5 (c), we conjecture that in general, the 'appropriate' range of λ is related to the magnitude of $|g'(x)|$, rather than independent of the target function class and training samples. Moreover, a multi-scale strategy that selects random parameters from various scopes can be beneficial, especially when the target function is complicated and composed of multiple 'modes.' In Figure 5 (d), we find another interesting result that the training output of the network with 300 hidden neurons and weights (and biases) randomly chosen from [-100,100] is not significantly affected if we remove 85 hidden neurons with weights (and biases) located in the 'narrow' range [-30,30]. It means, these hidden weights (and biases) as randomly assigned from [-30,30], not to mention the ones from [-1,1], provide a little contribution to approximation universality in learning.



(a) $\lambda = 1, L = 10000$

(b) $\lambda = 100, L = 300$

(c) $g'(x)$

(d) $y1$, $y2$, and $y1 - y2$

Figure 5. Performance for training results of $g(x)$: (a) $\lambda = 1, L = 10000$, (b) $\lambda = 100, L = 300$. (c) Derivative function $g'(x)$. (d) $\mathcal{N}_1$: Training approximation of $g(x)$ with hidden weights (and biases) randomly assigned from $[-100, 100]$, $\mathcal{N}_2$: Training approximation of $g(x)$ with hidden weights (and biases) randomly assigned from $[-100, 100]/[-30, 30]$, and their numerical difference $\mathcal{N}_1 - \mathcal{N}_2$.

Two main take-home messages: Our experiments support our theoretical results, which send two critical messages. (1) For a learning task, simply taking a fixed scope $[-\lambda, \lambda]$ would not make random neural nets universal approximators, if λ is not set properly. (2) For a gaussian-type target function $f(x;\sigma) = \exp(-|x|^2/\sigma^2)$ $(\sigma > 0)$, which is a Sobolev function and thus meets the conditions of our main theorem, a large value of λ is usually needed if σ is small. Our empirical findings provide valuable guidance for developing algorithms for constructing random neural networks. As a practical suggestion, users utilizing random networks for data modeling should be aware that the selection of the parameter λ greatly impacts the performance of the model. To determine an appropriate value of λ, it is recommended to conduct simulations through a trial-and-error approach. While this method is relatively straightforward to implement, it relies heavily on human intervention and is not a fully automated algorithm.

Our theoretical and empirical results indicate that randomly assigning weights from a fixed range or distribution that is independent of the training samples or prior knowledge may not be the most effective approach. Instead, it is more beneficial to explore different settings of random weights from various distributions, with the goal of expanding the function basis and increasing the ability to approximate the target function. Refer to the additional information provided in Appendix A. Further simulations on a 2D example and some real-world datasets are provided in Appendix D.1 and Appdendix D.2, respectively.

6. Conclusion and Discussion

In this paper, we examine the lower bound on the approximation error of shallow neural networks with random weights. Specifically, we explore the impact and limitations of randomness on the network's capacity for expression. Our theoretical findings indicate that the lower bound on the approximation error of a random network may not be zero if the range/distribution of hidden parameters is not appropriately selected in advance. Our results are based on the assumption of bandwidth limitation, which is a form of stochastic limitation that includes a finite variation, and are also valid when the proposed distribution is fully supported, such as a normal distribution.

Our theoretical results and empirical findings provide evidence that challenges the prevalent belief that a shallow random neural network is always a universal approximator regardless of the choice of hidden weights. This is significant as it helps researchers working with shallow neural networks and random weights to have a better understanding of the critical issues and potential drawbacks associated with randomness. Further in-depth analysis, both for deep neural networks or with tighter bounds, is expected to provide more insights. Interpretation of when and why neural networks with random weights are effective or not is essential to advance the understanding of this research topic.

Acknowledgements

Ming Li acknowledged the support from the National Natural Science Foundation of China (No. U21A20473, No. 62172370), and from the Zhejiang Provincial Natural Science Foundation (No. LY22F020004). Feilong Cao acknowledged the support from the National Natural Science Foundation of China (No. 62176244). Sho Sonoda was supported by JSPS KAKENHI 18K18113, JST CREST JPMJCR2015 and JST PRESTO JPMJPR2125. Yu Guang Wang acknowledged the support from the frontier research on the fundamentals of artificial intelligence mathematics (No. P22KN005). Jiye Liang acknowledged the support from the National Natural Science Foundation of China (No. U21A20473).

References

Bach, F. On the equivalence between kernel quadrature rules and random feature expansions. *Journal of Machine Learning Research*, 18(1):714–751, 2017.

Barbier, J., Macris, N., Dia, M., and Krzakala, F. Mutual information and optimality of approximate message-passing in random linear estimation. *IEEE Transactions on Information Theory*, 66(7):4270–4303, 2020.

Barron, A. R. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory*, 39(3):930–945, 1993.

Belkin, M., Hsu, D., Ma, S., and Mandal, S. Reconciling modern machine learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

Boutsidis, C., Zouzias, A., Mahoney, M. W., and Drineas, P. Randomized dimensionality reduction for k-means clustering. *IEEE Transactions on Information Theory*, 61 (2):1045–1062, 2014.

Candès, E. J. Harmonic analysis of neural networks. *Applied and Computational Harmonic Analysis*, 6(2):197–218, 1999.

Cao, W., Wang, X., Ming, Z., and Gao, J. A review on neural networks with random weights. *Neurocomputing*, 275:278–287, 2018.

Carroll, S. M. and Dickinson, B. W. Construction of neural nets using the Radon transform. In *International Joint Conference on Neural Networks 1989*, volume 1, pp. 607–611. IEEE, 1989.

Chen, C. P. and Liu, Z. Broad learning system: An effective and efficient incremental learning system without the need for deep architecture. *IEEE Transactions on Neural Networks and Learning Systems*, 29(1):10–24, 2017.

Cho, Y. and Saul, L. K. Kernel methods for deep learning. In *Advances in Neural Information Processing Systems*, pp. 342–350, 2009.

Daniely, A. SGD learns the conjugate kernel class of the network. In *Advances in Neural Information Processing Systems*, pp. 2422–2430, 2017.

Donoho, D. L. Emerging applications of geometric multiscale analysis. *Proceedings of the ICM, Beijing 2002*, pp. 209–233, 2002.

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

Funahashi, K.-I. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2 (3):183–192, jan 1989.

Ghorbani, B., Mei, S., Misiakiewicz, T., and Montanari, A. Limitations of lazy training of two-layers neural network. In *Advances in Neural Information Processing Systems*, pp. 9111–9121. 2019.

Giryes, R., Sapiro, G., and Bronstein, A. M. Deep neural networks with random gaussian weights : A universal classification strategy? *IEEE Transactions on Signal Processing*, 64(13):3444–3457, 2015.

Gnecco, G. A comparison between fixed-basis and variable-basis schemes for function approximation and functional optimization. *Journal of Applied Mathematics*, 2012, 2012.

Gorban, A. N., Tyukin, I. Y., Prokhorov, D. V., and Sofeikov, K. I. Approximation with random bases: Pro et contra. *Information Sciences*, 364:129–145, 2016.

Hsu, D., Sanford, C. H., Servedio, R., and Vlatakis-Gkaragkounis, E. V. On the approximation power of two-layer networks of random relus. In *Proceedings of the 34th Annual Conference on Learning Theory*, pp. 2423–2461. PMLR, 2021.

Huang, C., Li, M., Cao, F., Fujita, H., Li, Z., and Wu, X. Are graph convolutional networks with random weights feasible? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):2751–2768, 2023.

Igelnik, B. and Pao, Y.-H. Stochastic choice of basis functions in adaptive function approximation and the functional-link net. *IEEE Transactions on Neural Networks*, 6(6):1320–1329, 1995.

Irie, B. and Miyake, S. Capabilities of three-layered perceptrons. In *IEEE International Conference on Neural Networks*, pp. 641–648. IEEE, 1988.

Ito, Y. Representation of functions by superpositions of a step or sigmoid function and their applications to neural network theory. *Neural Networks*, 4(3):385–394, jan 1991.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, pp. 8571–8580, 2018.

Jaeger, H. Adaptive nonlinear system identification with echo state networks. In *Advances in Neural Information Processing Systems*, pp. 593–600, 2002.

Ji, Z., Telgarsky, M., and Xian, R. Neural tangent kernels, transportation mappings, and universal approximation. In *International Conference on Learning Representations*, 2020.

Johnson, W. B. and Lindenstrauss, J. Extensions of lipschitz mappings into a hilbert space. *Contemporary Mathematics*, 26:189–206, 1984.

Kainen, P. C., Kůrková, V., and Sanguineti, M. Approximating multivariable functions by feedforward neural nets. In Bianchini, M., Maggini, M., and Jain, L. C. (eds.), *Handbook on Neural Information Processing*, volume 49 of *Intelligent Systems Reference Library*, pp. 143–181. Springer Berlin Heidelberg, 2013.

Kleyko, D., Kheffache, M., Frady, E. P., Wiklund, U., and Osipov, E. Density encoding enables resource-efficient randomly connected neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(8):3777–3783, 2021.

Klusowski, J. M. and Barron, A. R. Approximation by Combinations of ReLU and Squared ReLU Ridge Functions with ℓ^1 and ℓ^0 Controls. *IEEE Transactions on Information Theory*, 64(12):7649–7656, 2018.

Kostadinova, S., Pilipović, S., Saneva, K., and Vindas, J. The ridgelet transform of distributions. *Integral Transforms and Special Functions*, 25(5):344–358, 2014. doi: 10.1080/10652469.2013.853057.

Kůrková, V. and Sanguineti, M. Comparison of worst case errors in linear and neural network approximation. *IEEE Transactions on Information Theory*, 48(1):264–275, 2002.

Lee, H., Ge, R., Ma, T., Risteski, A., and Arora, S. On the ability of neural nets to express distributions. In *Proceedings of 30th Annual Conference on Learning Theory*, pp. 1–26, 2017.

Li, M. and Wang, D. Insights into randomized algorithms for neural networks: Practical issues and common pitfalls. *Information Sciences*, 382:170–178, 2017.

Li, M. and Wang, D. 2-D stochastic configuration networks for image data analytics. *IEEE Transactions on Cybernetics*, 51(1):359–372, 2021.

Li, M., Huang, C., and Wang, D. Robust stochastic configuration networks with maximum correntropy criterion for uncertain data regression. *Information Sciences*, 473: 73–86, 2019.

Liu, F., Huang, X., Chen, Y., and Suykens, J. A. Random features for kernel approximation: A survey in algorithms, theory, and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Louart, C., Liao, Z., Couillet, R., et al. A random matrix approach to neural networks. *The Annals of Applied Probability*, 28(2):1190–1248, 2018.

LukoševičIus, M. and Jaeger, H. Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149, 2009.

Malach, E., Yehudai, G., Shalev-Shwartz, S., and Shamir, O. Proving the lottery ticket hypothesis: Pruning is all you need. In *Proceedings of 37th International Conference on Machine Learning*, pp. 6682–6691, 2020.

Mallat, S. *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Academic Press, 2009.

Minsky, M. and Papert, S. *Perceptrons*. MIT press, 1988.

Mongia, M., Kumar, K., Erraqabi, A., and Bengio, Y. On random weights for texture generation in one layer neural networks. *arXiv preprint arXiv:1612.06070*, 2016.

Murata, N. An integral representation of functions using three-layered networks and their approximation bounds. *Neural Networks*, 9(6):947–956, 1996.

Neal, R. M. *Bayesian Learning for Neural Networks*. Lecture Notes in Statistics. Springer-Verlag New York, 1996.

Ongie, G., Willett, R., Soudry, D., and Srebro, N. A function space view of bounded norm infinite width relu nets: The multivariate case. In *International Conference on Learning Representations*, 2020.

Pao, Y. H., Park, G. H., and Sobajic, D. J. Learning and generalization characteristics of the random vector functional-link net. *Neurocomputing*, 6(2):163–180, 1994.

Parhi, R. and Nowak, R. D. Banach space representer theorems for neural networks and ridge splines. *Journal of Machine Learning Research*, 22(43):1–40, 2021.

Pennington, J. and Worah, P. Nonlinear random matrix theory for deep learning. In *Advances in Neural Information Processing Systems*, pp. 2637–2646, 2017.

Rahimi, A. and Recht, B. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, pp. 1177–1184, 2008a.

Rahimi, A. and Recht, B. Uniform approximation of functions with random bases. In *Proceedings of the 46th Annual Allerton Conference on Communication, Control, and Computing*, pp. 555–561, 2008b.

Rahimi, A. and Recht, B. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *Advances in Neural Information Processing Systems*, pp. 1313–1320, 2009.

Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386, 1958.

Rubin, B. The Calderón reproducing formula, windowed X-ray transforms, and radon transforms in L^p-spaces. *Journal of Fourier Analysis and Applications*, 4(2):175–197, 1998.

Savarese, P., Evron, I., Soudry, D., and Srebro, N. How do infinite width bounded norm networks look in function space? In *Proceedings of the 32nd Annual Conference on Learning Theory*, pp. 2667–2690, 2019.

Saxe, A. M., Koh, P. W., Chen, Z., Bhand, M., Suresh, B., and Ng, A. Y. On random weights and unsupervised feature learning. In *Proceedings of the 28th International Conference on Machine Learning*, pp. 1089–1096, 2011.

Scardapane, S. and Wang, D. Randomness in neural networks: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 7(2):e1200, 2017.

Schmidt, W. F., Kraaijveld, M. A., and Duin, R. P. Feedforward neural networks with random weights. In *Proceedings of the 11th International Conference on Pattern Recognition*, pp. 1–4, 1992.

Sonoda, S. Fast approximation and estimation bounds of kernel quadrature for infinitely wide models. *arXiv preprint arXiv:1902.00648*, 2019.

Sonoda, S. and Murata, N. Sampling hidden parameters from oracle distribution. In *Proceedings of International Conference on Artificial Neural Networks*, pp. 539–546. Springer International Publishing, 2014.

Sonoda, S. and Murata, N. Neural network with unbounded activation functions is universal approximator. *Applied and Computational Harmonic Analysis*, 43(2):233–268, 2017.

Sonoda, S., Ishikawa, I., Ikeda, M., Hagihara, K., Sawano, Y., Matsubara, T., and Murata, N. The global optimum of shallow neural network is attained by ridgelet transform. *arXiv preprint arXiv:1805.07517*, 2018.

Sonoda, S., Ishikawa, I., and Ikeda, M. Ridge Regression with Over-Parametrized Two-Layer Networks Converge to Ridgelet Spectrum. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021*, volume 130, pp. 2674–2682. PMLR, 2021a.

Sonoda, S., Ishikawa, I., and Ikeda, M. Ghosts in neural networks: Existence, structure and role of infinite-dimensional null space. *arXiv preprint arXiv:2106.04770*, 2021b.

Starck, J.-L., Murtagh, F., and Fadili, J. M. The ridgelet and curvelet transforms. In *Sparse Image and Signal Processing: Wavelets, Curvelets, Morphological Diversity*, pp. 89–118. Cambridge University Press, 2010. doi: 10.1017/CBO9780511730344.006.

Suzuki, T. Fast generalization error bound of deep learning from a kernel perspective. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics*, pp. 1397–1406, 2018.

Wang, D. and Li, M. Robust stochastic configuration networks with kernel density estimation for uncertain data regression. *Information Sciences*, 412:210–222, 2017a.

Wang, D. and Li, M. Stochastic configuration networks: Fundamentals and algorithms. *IEEE Transactions on Cybernetics*, 47(10):3466–3479, 2017b.

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

Yehudai, G. and Shamir, O. On the power and limitations of random features for understanding neural networks. In *Advances in Neural Information Processing Systems*, pp. 6594–6604. 2019.

Zhang, B., Miller, D. J., and Wang, Y. Nonlinear system modeling with random matrices: echo state networks revisited. *IEEE Transactions on Neural Networks and Learning Systems*, 23(1):175–182, 2011.

A. Further Background

The initial motivation of this work comes from the comments posted by Yann LeCun[1], where some truth background behind randomness in neural nets was briefly revisited. We do not only agree with Yann's comments after we conduct a comprehensive literature review for this line of research but also technically question the feasibility and effectiveness of the "neural nets with random weights" (with certain controversial name/term), since many researchers found empirically that, in some cases, although not always, random models with inappropriate setting of the random parameters lead to unstable or poor results .

Overall, that motivates us to investigate two pressing, however, puzzling questions: (1) Can we guarantee that a random neural net model with hidden parameters chosen from a fixed range, for example, a trivial case obtained by letting $\lambda = 1$, is a universal approximator? (2) Given a target function f with specified complexity, what is the relationship between an appropriate setting of λ (that can lead to a universal approximator in the sense of probability) and the smoothness of f? Though we raise these questions, our intention is not to make any judgement on or get involved in the controversial name towards this direction. Instead, we present our current study along the right track of neural nets with random weights (or random neural nets, random nets), with particular concerns on the theoretical aspects, aiming to provide some new insights into answering the above questions.

The appearance of randomness in neural networks can trace back to the original Rosenblatt's perceptron (Rosenblatt, 1958), where the first layer is randomly connected and later Minsky and Papert's Gamba perceptron (Minsky & Papert, 1988) whose first layer is a bunch of linear threshold units. In early 1990s, researchers made random training methods/models reification, for example, with single hidden layer feedforward networks (SLFNs) with random weights (Schmidt et al., 1992) and random vector functional-link (RVFL) networks (Pao et al., 1994). Algorithmically, they performed the two steps (mentioned at the beginning of the introduction section) to build the randomized learner model. However, the approximation errors of the resulting models are bounded in the statistical sense (Igelnik & Pao, 1995), implying that preferable approximation performance is not guaranteed for every random assignment of the hidden parameters if the re-given probability distribution $Q(a; b)$ is not appropriately chosen (Gorban et al., 2016; Li & Wang, 2017). In contrast, the stochastic configuration networks (Wang & Li, 2017b) can ensure universal approximation by enforcing certain constraints on the random assignment of the hidden parameters, rather than using the purely random way as the "good" probability distribution $Q^*(a; b)$ is unknown and data-dependent. Sonoda & Murata (2014) proposed the sampling regression learning method by introducing a nonparametric probability distribution of the hidden parameters of SLFNs, and fitting the output parameters via ordinary linear regression. Kleyko et al. (2021) proposed to represent input features of RVFLs via density-based encoding, which is widely known in the area of stochastic computing, and used the operations of binding and bundling from the area of hyper-dimensional computing for obtaining the activations of the hidden neurons. The framework of a broad learning system (Chen & Liu, 2017) performs in the manner of a flat network, in which the original inputs are transferred and placed as the "mapped features" in feature nodes and the structure is expanded in a wide sense in the "enhancement nodes."

Although we only pay attention to shallow NNRWs with Step I and Step II (mentioned in the introduction), some other techniques/models using randomness are still worth mentioning here, aiming to present the engaging readers with a big picture of this line of research. For instance, the use of randomness in deep neural nets is also concerned in terms of different viewpoints. Mongia et al. (2016) showed that simple single-layer CNNs with random filters could serve as the basis for excellent texture synthesis models. Saxe et al. (2011) observed that the results of a learner based on random weights are comparable to that after regular pre-training and fine-tuning processes. Giryes et al. (2015) showed that under certain conditions, DNNs with random gaussian weights could perform a stable embedding of the original data, permitting a stable recovery of the data from the features represented by the network. Reservoir computing, a new paradigm to use recurrent neural networks with fixed and randomly generated weights, has also been widely adopted in-stream data modeling tasks (Jaeger, 2002; LukoševičIus & Jaeger, 2009; Zhang et al., 2011). Kernel approximation with random features (Rahimi & Recht, 2008a;b; 2009) can also be viewed as a random training method as its primary philosophy is mapping the input data to a randomized low-dimensional feature space and then applying existing fast linear methods. See the recent survey paper (Liu et al., 2021). On the other hand, random projections are well established and commonly used for dimensionality reduction (Boutsidis et al., 2014; Barbier et al., 2020). Here, one utilizes a random matrix to project input patterns from a high-dimensional space to a lower-dimensional representation such that distances between these patterns are preserved with high accuracy, as stated in Johnson-Lindenstrauss lemma (Johnson & Lindenstrauss, 1984).

[1] https://www.facebook.com/yann.lecun/posts/10152872571572143

B. Integral Representation Theory and Ridgelet Transform

B.1. Background

The *ridgelet transform* has been independently discovered by Murata (1996), Candès (1999), and Rubin (1998) during 1996–1998 as a 'harmonic analysis of neural networks'. This is a path-breaking study, not only in the neural network field, but also in the sparse coding theory (see overviews by Donoho (2002) and by Starck et al. (2010)). The ridgelet transform has been extended to Schwartz distributions by Kostadinova et al. (2014), and to non-integrable activation functions such as ReLU by Sonoda & Murata (2017). The *integral representation of a neural network* had been developed before the ridgelet transform. (Recall that the ridgelet transform R is a right inverse operator of the integral representation operator S. Thus, we can analyze S without knowing R.) For example, Irie & Miyake (1988), Funahashi (1989) and Barron (1993) used Fourier transform as an integral representation to prove the UAP. Carroll & Dickinson (1989) and Ito (1991) used Radon transform. In particular, the so-called *Barron class* (proposed in Barron (1993)) characterizes the functions that neural networks can effectively approximate. The effectiveness here is quantified as *Barron's bound*, a dimension-free approximation upper bound (see the overview by Kainen et al. (2013)). The original Barron's theory excludes ReLU, and the upper bound is in general not tight. Thus, many authors (Klusowski & Barron, 2018; Lee et al., 2017; Sonoda, 2019; E et al., 2019; Savarese et al., 2019; Ji et al., 2020; Ongie et al., 2020; Parhi & Nowak, 2021) have improved and developed Barron-like theories for ReLU nets. It is notable that Ongie et al. (2020) and Parhi & Nowak (2021) have employed the Radon transform and developed some representer theorems. The novelty of this study in the integral representation literature is in the estimation of lower bounds.

B.2. Quick Overview

We explain the integral representation theory established in (Sonoda & Murata, 2017) showing a few new results. In order to avoid confusion, we use two symbols $\hat{\cdot}$ and $\cdot^\sharp$ for m-dimensional and 1-dimensional Fourier transforms respectively, namely,

$$\widehat{f}(\xi) := \int_{\mathbb{R}^m} f(x) e^{-i\xi\cdot x} \mathrm{d}x, \quad f \in L^2(\mathbb{R}^m), \xi \in \mathbb{R}^m;$$
$$\sigma^\sharp(\omega) := \int_{\mathbb{R}} \sigma(t) e^{-i\omega t} \mathrm{d}t, \quad \sigma \in L^2(\mathbb{R}), \omega \in \mathbb{R}.$$

Let P be a Radon measure on $\mathbb{R}^m$. We consider two Hilbert spaces $\mathcal{F} = L^2(P)$ and $\mathcal{G} = L^2(\mathbb{R}^m \times \mathbb{R})$ associated with the following inner products:

$$\langle f, g \rangle_{\mathcal{F}} := \int_{\mathbb{R}^m} f(x)\overline{g(x)} \mathrm{d}P(x),$$
$$\langle \phi, \psi \rangle_{\mathcal{G}} := \int_{\mathbb{R}^m \times \mathbb{R}} \phi(a,b)\overline{\psi(a,b)} \mathrm{d}a\mathrm{d}b,$$

and the Banach space $\mathcal{M}$ of the finite Radon measures on $\mathbb{R}^m \times \mathbb{R}$ equipped with the total variation norm $\|\cdot\|_{TV}$.

Definition 4 (Integral representation S)**.** Fix any function $\sigma : \mathbb{R} \to \mathbb{C}$ and measure μ on $\mathbb{R}^m \times \mathbb{R}$, we define the integral representation as

$$S[\mu](x) := \int_{\mathbb{R}^m \times \mathbb{R}} \sigma(a \cdot x - b) \mathrm{d}\mu(a,b), \quad x \in \mathbb{R}^m.$$

With a slight abuse of notation, when the measure μ has a density $\phi \in L^1(\mathbb{R}^m \times \mathbb{R})$, we write $S[\phi]$ instead of $S[\phi \mathrm{d}a\mathrm{d}b]$.

Proposition 3 (Fourier expression of S)**.**

$$S[\mu](x) = \frac{1}{2\pi} \int_{\mathbb{R}^m \times \mathbb{R}} \mu^\sharp(a,\omega)\sigma^\sharp(\omega) \mathrm{d}\omega e^{i\omega a\cdot x} \mathrm{d}a.$$

Proof. Since $\sigma(a \cdot x - b) = \frac{1}{2\pi}\int_{\mathbb{R}} \sigma^\sharp(\omega) e^{i\omega(a\cdot x - b)} \mathrm{d}\omega$,

$$S[\mu](x) = \frac{1}{2\pi} \int_{\mathbb{R}^m \times \mathbb{R}} \sigma^\sharp(\omega) e^{i\omega(a\cdot x-b)} \mathrm{d}\omega \mathrm{d}\mu(a,b)$$
$$= \frac{1}{2\pi} \int_{\mathbb{R}^m \times \mathbb{R}} \mu^\sharp(a,\omega)\sigma^\sharp(\omega) e^{i\omega a\cdot x} \mathrm{d}\omega \mathrm{d}a. \qquad \square$$

Proposition 4 (Boundedness (Lipschitz continuity) of $S : \mathcal{M} \to \mathcal{F}$)**.** *We write $\sigma_{a,b}(x) := \sigma(a \cdot x - b)$. Provided that the constant $C_{\sigma,P}^2 := \sup_{a,b} \|\sigma_{a,b}\|_{\mathcal{F}}^2$ exists finite, then $S : \mathcal{M} \to \mathcal{F}$ is bounded, or equivalently, Lipschitz continuous: For any $\mu \in \mathcal{M}$, we have*

$$\|S[\mu]\|_{\mathcal{F}}^2 \leq \int \left(\int |\sigma_{a,b}(x)| \mathrm{d}|\mu|(a,b) \right)^2 \mathrm{d}P(x) \leq \|\mu\|_{TV}^2 \int \int |\sigma_{a,b}(x)|^2 \frac{\mathrm{d}|\mu|(a,b)}{\|\mu\|_{TV}} \mathrm{d}P(x) \leq C_{\sigma,P}^2 \|\mu\|_{TV}^2.$$

The boundedness of S is a sufficient condition for the optimization problem to be well-defined, in the sense that $S(\mathcal{M}) \subset \mathcal{F}$. Hence, unless otherwise noted, we always assume that $C_{\sigma,P} < \infty$.

Proposition 5 (Adjoint operator S_P^*)**.** *For $S : \mathcal{G} \to \mathcal{F}$, the adjoint operator $S_P^* : \mathcal{F} \to \mathcal{G}$ is given by*

$$S_P^*[f](a,b) = \int_{\mathbb{R}^m} f(x) \overline{\sigma(a \cdot x - b)} \mathrm{d}P(x).$$

If P is obvious from the context, we write S_P^* as S^* for simplicity.

Proof. We can verify this by the direct calculation: For any $f \in \mathcal{F}$ and $\phi \in \mathcal{G}$,

$$\langle f, S[\phi] \rangle_{\mathcal{F}} = \int_{\mathbb{R}^m \times \mathbb{R} \times \mathbb{R}^m} f(x) \overline{\sigma(a \cdot x - b)\phi(a,b)} \mathrm{d}a \mathrm{d}b \mathrm{d}P(x) = \langle S_P^*[f], \phi \rangle_{\mathcal{G}},$$

as long as one of the integrals exists. □

Definition 5 (Ridgelet transform R)**.** For any measure P on $\mathbb{R}^m$ and function $\rho : \mathbb{R} \to \mathbb{C}$, we define the *ridgelet transform* of f on $\mathbb{R}^m$ by

$$R_P[f;\rho](a,b) := \int_{\mathbb{R}^m} f(x) \overline{\rho(a \cdot x - b)} \mathrm{d}P(x), (a,b) \in \mathbb{R}^m \times \mathbb{R}.$$

If P and/or ρ are obvious from the context, we write $R_P[f;\rho]$ as $R[f]$ for simplicity. In addition, when we emphasize the Lebesgue measure case $P = \mathrm{d}x$, we write $R_{\mathrm{d}x}$.

In particular, the adjoint operator S_P^* is a ridgelet transform: $S_P^*[f] = R_P[f;\sigma]$.

Proposition 6 (Fourier expression of R)**.**

$$R_P[f;\rho](a,b) = \frac{1}{2\pi} \int_{\mathbb{R}} \widehat{f\mathrm{d}P}(\omega a) \overline{\rho^\sharp(\omega)} e^{i\omega b} \mathrm{d}\omega,$$

where $\widehat{f\mathrm{d}P}$ denotes the Fourier transform of the measure $f\mathrm{d}P$. When $P = \mathrm{d}x$, then $\widehat{f\mathrm{d}x}$ is naturally identified with the ordinary Fourier transform $\widehat{f}$.

Proof. Since $\rho(a \cdot x - b) = \frac{1}{2\pi} \int_{\mathbb{R}} \rho^\sharp(\omega) e^{i\omega(a \cdot x - b)} \mathrm{d}\omega$,

$$\begin{aligned} R_P[f;\rho](a,b) &= \frac{1}{2\pi} \int_{\mathbb{R}^m} f(x) \int_{\mathbb{R}} \overline{\rho^\sharp(\omega)} e^{-i\omega(a \cdot x - b)} \mathrm{d}\omega \mathrm{d}P(x) \\ &= \frac{1}{2\pi} \int_{\mathbb{R}} \widehat{f\mathrm{d}P}(\omega a) \overline{\rho^\sharp(\omega)} e^{i\omega b} \mathrm{d}\omega. \end{aligned}$$

□

We remark that satisfying this admissible condition is not difficult. For example, take a Gaussian $\rho_0(t) := \exp(-t^2/2)$, and put $\rho_n(t) := C\rho_0^{(n)}(t)$ with an integer n such that $2n - m > 0$ and a positive constant C. Then, by appropriately setting C, ρ_n can be admissible (with itself) because $(2\pi)^{m-1} \int_{\mathbb{R}} |\rho_n^\sharp(\omega)|^2 / |\omega|^m \mathrm{d}\omega = (2\pi)^{m-1} C \int_{\mathbb{R}} |\omega|^{2n-m} |\rho_0^\sharp(\omega)|^2 \mathrm{d}\omega < \infty$ and we can set C for the integral to be normalized as 1.

Proposition 7 (Reconstruction formula)**.** *For any $f \in L^1(P)$, $S[R_P[f;\rho]] = f\mathrm{d}P$.*

Proof. We write $f_P := f\mathrm{d}P$ for short. By using the Fourier expressions, we have

$$\begin{aligned} S[R_P[f;\rho]](x) &= \frac{1}{2\pi}\int_{\mathbb{R}^m\times\mathbb{R}} \widehat{f_P}(\omega a)\sigma^\sharp(\omega)\overline{\rho^\sharp(\omega)}e^{i\omega a\cdot x}\mathrm{d}\omega\mathrm{d}a \\ &= \frac{1}{2\pi}\int_{\mathbb{R}^m} \widehat{f_P}(\xi)\left[\int_{\mathbb{R}} \frac{\sigma^\sharp(\omega)\overline{\rho^\sharp(\omega)}}{|\omega|^m}\mathrm{d}\omega\right] e^{i\xi\cdot x}\mathrm{d}\xi \\ &= f_P(x). \end{aligned}$$

Here, we change the variable $(a,\omega) = (\xi/\omega, \omega)$ with $\mathrm{d}a\mathrm{d}\omega = |\omega|^{-m}\mathrm{d}\xi\mathrm{d}\omega$ in the second equation. □

We remark that when σ is self-admissible, the reconstruction formula can be extended to $f \in L^2(\mathbb{R}^m)$ by using the Plancherel formula below.

The following isometries play an important role in the proof of main results as we can regard $|R[f](a,b)|^2$ with a "density function" of the parameter distribution.

Proposition 8 (Plancherel formula)**.**

- *When* $\sigma = \rho$, $\|S_P^*[f]\|_{\mathcal{G}}^2 = \langle f, f\mathrm{d}P\rangle_{\mathcal{F}}$ *because*

$$\|S_P^*[f]\|_{\mathcal{G}}^2 = \langle S_P^*[f], S_P^*[f]\rangle_{\mathcal{G}} = \langle f, S[S_P^*[f]]\rangle_{\mathcal{F}} = \langle f, f\mathrm{d}P\rangle_{\mathcal{F}}.$$

- *When* $\sigma = \rho$, $\|f\|_{\mathcal{F}}^2 = \langle S_P^*[f], S_{\mathrm{d}x}^*[f]\rangle_{\mathcal{G}}$ *because*

$$\|f\|_{\mathcal{F}}^2 = \langle f, f\rangle_{\mathcal{F}} = \langle f, S[S_{\mathrm{d}x}^*[f]]\rangle_{\mathcal{F}} = \langle S_P^*[f], S_{\mathrm{d}x}^*[f]\rangle_{\mathcal{G}}.$$

- *When* f *is supported in a set* $\mathcal{X} \subset \mathbb{R}^m$ *and* $P = 1_{\mathcal{X}}\mathrm{d}x$ *(indicator function), then* $f\mathrm{d}P = f$*, and thus* $S_{\mathrm{d}x}^*[f] = S_P^*[f]$*, and the above two identities coincide:* $\|S_P^*[f]\|_{\mathcal{G}}^2 = \|f\|_{\mathcal{F}}^2$.

C. Proofs for Theorems

C.1. Theorem 1

We impose assumptions as below.

(A1) Let Ω be a bounded open subset with smooth boundary in the input domain $\mathbb{R}^m$, and put $K := \overline{\Omega}$. Namely, K is a compact set. The boundedness assumption is required for the loss $\|f - g_d\|_{L^2(K)}$ between f and *finite net* $g_d(x) = \sum_{i=1}^d c_i\sigma(a_i \cdot x - b_i)$ exists finite. We note that $\|g_d\|_{L^2(\mathbb{R}^m)} = \infty$ simply because $\sigma(a\cdot x - b)$ has a constant direction, while $\|g_d\|_{L^2(K)} < \infty$. The closure assumption excludes degenerated cases such as $K = \{x_1, \ldots, x_n\}$ (isolated points) for the sake of simplicity. The smooth boundary is required in Theorem 2, to continuously embed $H^s(\Omega)$ to $H^s(\mathbb{R}^m)$ via zero-extension.

(A2) Let $f : \mathbb{R}^m \to \mathbb{C}$ be an square-integrable function supported in the compact set K. Namely, $f \in L^2(K)$. Both integrability and compact-support assumptions exclude the so-called "teacher-student setting" where f is a finite neural network such as $\sum_{i=1}^d c_i\sigma(a_i \cdot x - b_i)$.

(A3) $P := 1_K\mathrm{d}x$ (i.e., the volume is not normalized to 1), which yields $\mathcal{F} = L^2(P) = L^2(K)$ and $S^*[f] = R[f;\sigma,1_K\mathrm{d}x] = S_K^*[f]$.

(A4) $C_{\sigma,P}$ exists finite, namely $\|S[\mu]\|_{\mathcal{F}} \leq C_{\sigma,P}\|\mu\|_{TV}$ so that $S(\mathcal{M}) \subset \mathcal{F}$.

(A5) Let $\sigma : \mathbb{R} \to \mathbb{C}$ be a measurable function that is admissible with itself.

Then, the approximation error is lower bounded by the volume of the tail part (i.e., outside the parameter domain V) of the ridgelet spectrum:

$$\inf_{\mu\in\mathcal{M}(V)} \|f - S[\mu]\|^2_{L^2(K)} \geq \|f\|^2_{L^2(K)} - \|S[S^*_K[f]|_V]\|^2_{L^2(K)} \geq \|f\|^2_{L^2(K)} - \|S^*_K[f]\|^2_{L^2(V)} = \|S^*_K[f]\|^2_{L^2(V^c)}.$$

Obviously, the lower bound is strictly positive when the tail density $S^*[f]|_{V^c}$ is positive.

Proof. We write $\mathcal{G} := L^2(\mathbb{R}^m \times \mathbb{R})$ for short. By $S^*[f]|_V$ (resp. $S^*[f]|_{V^c} = S^*[f] - S^*[f]|_V$) we write the truncation of the ridgelet spectum $S^*[f]$ onto V (resp. V^c). By $\mathrm{proj}_{\ker S}$ (resp. $\mathrm{proj}_{(\ker S)^\perp}$) we write the projection from $\mathcal{M}(V)$ to the null space $\ker S$ (resp. to its complement $(\ker S)^\perp$).

Step 1. Let μ^* denote an arbitrary single element of the minimizers in $\mathcal{M}(V)$. We note that μ^* always exists as a consequence of the following *extreme value theorem*:

Proposition 9. *Suppose E be a Banach space, X be a closed convex subset of E, and $\varphi : X \to (\infty, \infty]$ be a coercive lower semi-continuous function. (Here, coercive means $\varphi(x) \to +\infty$ as $\|x\|_E \to \infty$.) Then, there exists an element (minimizer) $x^* \in X$ that attains the minimum, i.e., $\inf_{x\in X} \varphi(x) = \varphi(x^*)$.*

Now $E = X = \mathcal{M}(V)$ is a Banach space (known as an *rca space*), which means it is closed and convex, and $\varphi(\mu) := \|f - S[\mu]\|^2_{L^2(K)}$ is coercive and Lipschitz continuous, there exists a minimizer $\mu^* \in \mathcal{M}(V)$. Namely, we have

$$\inf_{\mu\in\mathcal{M}(V)} \|f - S[\mu]\|^2_{L^2(K)} = \|f - S[\mu^*]\|^2_{L^2(K)}. \tag{5}$$

Since the minimizer $S[\mu^*]$ satisfies the Pythagorean relation:

$$\|S[\mu^*]\|^2_{L^2(K)} + \|f - S[\mu^*]\|^2_{L^2(K)} = \|f\|^2_{L^2(K)}, \tag{6}$$

we have

$$(5) = \|f\|^2_{L^2(K)} - \|S[\mu^*]\|^2_{L^2(K)}. \tag{7}$$

Step 2. We show the following inequality:

$$\|S[\mu^*]\|_{L^2(K)} \leq \|S[S^*[f]|_V]\|_{L^2(K)}, \tag{8}$$

which yields the following lower bound:

$$(7) \geq \|f\|^2_{L^2(K)} - \|S[S^*[f]|_V]\|^2_{L^2(K)} \tag{9}$$

Proof of (8). To estimate the norm of $S[\mu^*]$, we can neglect the null component of μ^*, say $\mu^*_0 \in \ker(S : \mathcal{M}(V) \to L^2(K))$, since it satisfies

$$\|S[\mu^*]\|_{L^2(K)} = \|S[\mu^* - \mu^*_0]\|_{L^2(K)}, \quad \text{and} \quad \langle f, S[\mu^*]\rangle_{L^2(K)} = \langle S^*_K[f], \mu^* - \mu^*_0\rangle_{L^2(\mathbb{R}^m\times\mathbb{R})}, \tag{10}$$

for any $f \in L^2(K)$. The Pythagorean relation (6) is rephrased as

$$\|S[\mu^*]\|^2_{L^2(K)} = \Re\langle f, S[\mu^*]\rangle_{L^2(K)}. \tag{11}$$

Since μ^* is supported in V,

$$= \Re\langle S^*_K[f]|_V, \mu^*\rangle_{L^2(\mathbb{R}^m\times\mathbb{R})}. \tag{12}$$

Since μ^* is assumed not to contain the null component,

$$= \Re\langle \mathrm{proj}_{(\ker S)^\perp}[S^*_K[f]|_V], S^*_K[S[\mu^*]]\rangle_{L^2(\mathbb{R}^m\times\mathbb{R})}, \tag{13}$$

By the definition of S_K^*,

$$= \Re \langle S[S^*[f]|_V], S[\mu^*] \rangle_{L^2(K)}. \tag{14}$$

By the Cauchy-Schwartz inequality,

$$\leq \| S[S^*[f]|_V] \|_{L^2(K)} \| S[\mu^*] \|_{L^2(K)}, \tag{15}$$

which yields the inequality (8). □

Step 3. Next, we show the following equalities:

$$\begin{aligned} \|f\|_{L^2(K)}^2 &= \|S_K^*[f]\|_{L^2(V)}^2 + \|S_K^*[f]\|_{L^2(V^c)}^2 \\ &= \|S[S_K^*[f]|_V]\|_{L^2(K)}^2 + \|S[S_K^*[f]|_{V^c}]\|_{L^2(K)}^2 + 2\|\phi_0\|_{L^2(\mathbb{R}^m \times \mathbb{R})}^2, \end{aligned} \tag{16}$$

where ϕ_0 is the null component of $S_K^*[f]|_V$ defined later, and this equality refines the lower bound as

$$(9) = \|S[S_K^*[f]|_{V^c}]\|_{L^2(K)}^2 + 2\|\phi_0\|_{L^2(\mathbb{R}^m \times \mathbb{R})}^2 \geq \|S^*[f]\|_{L^2(V^c)}^2 = \|f\|_{L^2(K)}^2 - \|S^*[f]\|_{L^2(V)}^2.$$

Proof of (16). By using the Plancherel formula and splitting the integral, the restrictions $S^*[f]|_V$ and $S^*[f]|_{V^c}(= S^*[f] - S^*[f]|_V)$ of ridgelet spectra $S^*[f]$ satisfy the following equation:

$$\begin{aligned} \|f\|_{L^2(K)}^2 &= \|S_K^*[f]\|_{L^2(\mathbb{R}^m \times \mathbb{R})}^2 \\ &= \left(\int_V + \int_{V^c} \right) |S_K^*[f](a,b)|^2 \mathrm{d}a\mathrm{d}b = \|S_K^*[f]\|_{L^2(V)}^2 + \|S_K^*[f]\|_{L^2(V^c)}^2. \end{aligned} \tag{17}$$

In order to further decompose the equation (17), we consider the null components of the restrictions $S_K^*[f]|_V$ and $S_K^*[f]|_{V^c}$. Recall that the operator $S : L^2(\mathbb{R}^m \times \mathbb{R}) \to L^2(K)$ has a non-trivial null space $\ker(S : L^2(\mathbb{R}^m \times \mathbb{R}) \to L^2(K))$, and its orthogonal complement is given by the image space $\operatorname{image} S_K^*$ of the adjoint operator $S_K^* : L^2(K) \to L^2(\mathbb{R}^m \times \mathbb{R})$, namely, $(\ker S)^\perp = \operatorname{image} S_K^*$. Hence, the entire space $\mathcal{G} := L^2(\mathbb{R}^m \times \mathbb{R})$ is decomposed into the orthogonal direct sum: $\mathcal{G} = \ker S \oplus \operatorname{image} S_K^*$. By definition, $S_K^*[f] \in \operatorname{image} S_K^* = (\ker S)^\perp$. Nevertheless, its restrictions $S_K^*[f]|_V$ and $S_K^*[f]|_{V^c}$ may have null components. We write the (potentially non-trivial) null component of S_K^* as $\phi_0 := \operatorname{proj}_{\ker S}[S_K^*[f]|_V]$, and its orthogonal component as $\phi_V := \operatorname{proj}_{(\ker S)^\perp}[S_K^*[f]|_V]$, so that both components become a direct sum: $\phi_V \oplus \phi_0 = S_K^*[f]|_V$. Then, the null component of $S_K^*[f]|_{V^c}$ is $-\phi_0$ because the sum $S_K^*[f]|_V + S_K^*[f]|_{V^c} = S_K^*[f]$ is in the image space $\operatorname{image} S_K^*$, and thus the orthogonal component $\phi_{V^c} := \operatorname{proj}_{(\ker S)^\perp}[S_K^*[f]|_{V^c}]$ is given by $\phi_{V^c} = S_K^*[f]|_{V^c} + \phi_0$. Hence, by using the orthogonality and the Plancherel formula, the equation (17) is further calculated as follows:

$$\begin{aligned} (17) &= \|\phi_V\|_{L^2(\mathbb{R}^m \times \mathbb{R})}^2 + \|\phi_{V^c}\|_{L^2(\mathbb{R}^m \times \mathbb{R})}^2 + 2\|\phi_0\|_{L^2(\mathbb{R}^m \times \mathbb{R})}^2 \\ &= \|S[S_K^*[f]|_V]\|_{L^2(K)}^2 + \|S[S_K^*[f]|_{V^c}]\|_{L^2(K)}^2 + 2\|\phi_0\|_{L^2(\mathbb{R}^m \times \mathbb{R})}^2. \end{aligned}$$

Combining Steps 1, 2, and 3, we have the assertion. □

C.2. Theorem 2

We write $\langle x \rangle := (1 + |x|^2)^{1/2}$ for $x \in \mathbb{R}^m$, which satisfies $\max\{1, |x|\} \leq \langle x \rangle$ for any x. For square-integrable functions $f \in L^2(\mathbb{R}^m)$ on whole space $\mathbb{R}^m$, we employ $\|f\|_{H^s(\mathbb{R}^m)}^2 := \int_{\mathbb{R}^m} |\widehat{f}(\xi)|^2 (1 + |\xi|^2)^s \mathrm{d}\xi$ for the L^2-Sobolev norm of order $s \in \mathbb{R}$. For functions on an open subset Ω with C^1-boundary, we define the L^2-Sobolev space $H^s(\Omega)$ with $s \in (1/2, \infty]$ by continuously embedding it to $H^s(\mathbb{R}^m)$. Namely, we identify $f \in H^s(\Omega)$ with $\overline{f} \in H^s(\mathbb{R}^m)$ that is compactly supported in Ω and satisfies $\overline{f}|_\Omega = f$.

Decay property. Suppose that ρ is self-admissible, namely, $\int_{\mathbb{R}} |\rho^\sharp(\omega)|^2 |\omega|^{-m} \mathrm{d}\omega = (2\pi)^{m-1}$. For any $f \in H^s(\Omega)$,

$$\begin{aligned} |R[f](ru,b)| &\leq \frac{1}{2\pi} \int_{\mathbb{R}} |\widehat{f}(\omega u)| |\rho^\sharp(\omega/r)/r| \mathrm{d}\omega \\ &\leq \frac{1}{2\pi} \int_{\mathbb{R}} \left(|\omega u|^s |\widehat{f}(\omega u) \omega^{\frac{m-1}{2}}| \right) \left(|\omega^{-(2s+m-1)/2} \rho^\sharp(\omega/r)/r| \right) \mathrm{d}\omega \\ &\leq \frac{1}{2\pi} \left(2 \int_0^\infty \langle \omega u \rangle^{2s} |\widehat{f}(\omega u)|^2 \omega^{m-1} \mathrm{d}\omega \right)^{1/2} \left(|r|^{-2s-m} \int_{\mathbb{R}} \langle \omega \rangle^{-2s+1} |\rho^\sharp(\omega)|^2 |\omega|^{-m} \mathrm{d}\omega \right)^{\frac{1}{2}} \\ &= C_{\rho,s} |r|^{-(2s+m)/2} \Phi_s[f](u). \end{aligned}$$

Here, we write $\Phi_s[f](u) := \left(\int_0^\infty \langle \omega u \rangle^{2s} |\widehat{f}(\omega u)|^2 \omega^{m-1} \mathrm{d}\omega \right)^{1/2}$ for future use, of which the spherical mean becomes the Sobolev norm:

$$\int_{\mathbb{S}^{m-1}} \Phi_s[f](u)^2 \mathrm{d}u = \|f\|_{H^s}^2;$$

and the constant $C_{\rho,s}$ is given and bounded as

$$C_{\rho,s}^2 := \frac{2}{(2\pi)^2} \int_{\mathbb{R}} \langle \omega \rangle^{-2s+1} \frac{|\rho^\sharp(\omega)|^2}{|\omega|^m} \mathrm{d}\omega \le \frac{2}{(2\pi)^2} \int_{\mathbb{R}} \frac{|\rho^\sharp(\omega)|^2}{|\omega|^m} \mathrm{d}\omega = 2(2\pi)^{m-3} < \infty,$$

because $\langle \omega \rangle^{-2s+1} \le 1$ as long as $-2s+1 \le 1$.

Auxiliary estimates. The obtained estimate does not depend on b and diverges at $r = 0$, but $R[f]$ usually depends on b and does not always diverge at $r = 0$. Hence, we derive auxiliary estimates. By the assumption that $f \in L^1(K)$ and $\rho \in L^\infty(\mathbb{R})$, we have

$$|R[f](a,b)| \le \int_{\mathbb{R}^m} |f(x)||\rho(a \cdot x - b)| \mathrm{d}x \le \|f\|_{L^1(K)} \|\rho(a \cdot x - b)\|_{L^\infty(K)} \le \|f\|_{L^1(K)} \|\rho\|_{L^\infty(\mathbb{R})}.$$

Therefore, put

$$\phi_a(ru) := \min\left\{ \|\rho\|_{L^\infty(\mathbb{R})} \|f\|_{L^1(K)}, C_{\rho,s} \Phi_s[f](u) r^{\frac{-2s-m}{2}} \right\}, \quad \phi_b(b) := \frac{|R[f](ru,b)|}{\phi_a(ru)}.$$

Since the estimate $|R[f](ru,b)| \le \phi_a(ru)$ is independent of b, ϕ_b is well-defined and uniformly bounded as $|\phi_b| \le 1$. By the square integrability of $R[f]$, we can decompose the integral as

$$\|R[f]\|_{L^2(\mathbb{R}^m \times \mathbb{R})}^2 = \int_{\mathbb{R}^m} |\phi_a(a)|^2 \mathrm{d}a \int_{\mathbb{R}} |\phi_b(b)|^2 \mathrm{d}b.$$

C.3. Theorem 3

Proof. We write $C_0 := \|\sigma\|_{L^\infty(\mathbb{R})} \|f\|_{L^1(K)}$ and $C_\infty := C_{\sigma,s} \|f\|_{H^s(\Omega)}$ for short. Let $V_a := \{a \in \mathbb{R}^m \mid |a| \le \lambda\}$ and $V_b := \{b \in \mathbb{R} \mid |b| \le \kappa\}$ so that $V = V_a \times V_b$. By Theorem 1, the approximation error $\inf_{\mu \in \mathcal{M}(p)} \|f - S[\mu]\|_{L^2(K)}^2$ is lower bounded by the tail bound $\|S_K^*[f]\|_{L^2(V^c)}^2 = \|f\|_{L^2(K)}^2 - \|S^*[f]\|_{L^2(V)}^2$. On the other hand, by Theorem 2, the parameter "density" $|S_K^*[f]|^2$ is upper bounded by a dominating function $|\phi_a|^2$; Furthermore, the integration of $|S_K^*[f]|^2$ over a product space $V_a \times V_b$ is exactly decomposed into the integrations of $|\phi_a|^2$ and $|\phi_b|^2$. In the following, by integrating the dominating function over the bandlimited domain V, we estimate the tail bound.

We begin with decomposing the integral as

$$\|S_K^*[f]\|_{L^2(V)}^2 = \left(\int_{\mathbb{S}^{m-1}} \int_0^\lambda |\phi_a(ru)|^2 r^{m-1} \mathrm{d}r \mathrm{d}u \right) \int_{V_b} |\phi_b(b)|^2 \mathrm{d}b = \|\phi_a\|_{L^2(V_a)}^2 \|\phi_b\|_{L^2(V_b)}^2.$$

Thus, we compute $\|\phi_a\|_{L^2(V_a)}^2$ in the following.

By averaging ϕ_a in direction $u \in \mathbb{S}^{m-1}$,

$$\int_{\mathbb{S}^{m-1}} |\phi_a(ru)|^2 \mathrm{d}u = \min\{C_0^2 \Omega_{m-1}, C_\infty^2 r^{-2s-m}\}.$$

Here, $\Omega_{m-1} = 2\pi^{m-1}/\Gamma(m/2)$ is the surface area of $\mathbb{S}^{m-1}$. Therefore, the rate in r changes at the cross point $r = \vartheta$ satisfying $C_0^2 \Omega_{m-1} = C_\infty^2 \vartheta^{-2s-m}$.

Let us consider the case $\lambda \le \vartheta$. Then,

$$\|\phi_a\|_{L^2(V_a)}^2 = \left(\int_{\mathbb{S}^{m-1}} \int_0^\lambda |\phi_a(ru)|^2 r^{m-1} \mathrm{d}r \mathrm{d}u \right) = C_0^2 V_m \lambda^m =: I_0(\lambda).$$

Here, $V_m = \pi^{m/2}/\Gamma(m/2+1)$ is the volume of m-unit ball, and we used the relation $\Omega_{m-1}/m = V_m$. Next, let us consider the case $\lambda \geq \vartheta$.

$$\begin{aligned}
\|\phi_a\|^2_{L^2(V_a)} &= I_0(\vartheta) + \int_{\mathbb{S}^{m-1}} \int_{\vartheta}^{\lambda} |\phi_a(ru)|^2 r^{m-1} \mathrm{d}r\mathrm{d}u \\
&= C_0^2 V_m \vartheta^m - \frac{C_\infty^2}{2s}\left(\lambda^{-2s} - \vartheta^{-2s}\right) \\
&= \frac{C_\infty^2}{2s}\left(-\lambda^{-2s} + \frac{2s+m}{m}\vartheta^{-2s}\right),
\end{aligned}$$

where the final equation is immediate from the relation $mC_0^2 V_m \vartheta^m = C_\infty^2 \vartheta^{-2s}$. By the positivity of integrand $|\phi_a|^2$, the final estimate is also positive (inspite of the negative term $-\lambda^{-2s}$).

As a biproduct, by letting $\lambda \to \infty$, we can verify that both ϕ_a and ϕ_b are finite measures on $\mathbb{R}^m$ and $\mathbb{R}$ respectively:

$$\begin{aligned}
&\|\phi_a\|^2_{L^2(\mathbb{R}^m)} = C_\infty^2 \left(\frac{2s+m}{2sm}\right)\vartheta^{-2s} \in (0,\infty), \\
\implies &\|\phi_b\|^2_{L^2(V_b)} = \|S^*[f]\|^2_{L^2(\mathbb{R}^m \times V_b)} / \|\phi_a\|^2_{L^2(\mathbb{R}^m)} \in (0,\infty).
\end{aligned}$$

To conclude, we have the following approximation lower bound:

$$\begin{aligned}
&\inf_{\mu \in \mathcal{M}(d)} \|f - S[\mu]\|^2_{L^2(K)} \\
&\geq \inf_{\mu \in \mathcal{M}(V)} \|f - S[\mu]\|^2_{L^2(K)} \\
&\geq \|S_K^*[f]\|^2_{L^2(V^c)} \\
&= \|f\|^2_{L^2(K)} - \|S^*[f]\|^2_{L^2(V)} \\
&= \|f\|^2_{L^2(K)} - \|\phi_b\|^2_{L^2(V_b)} \cdot \begin{cases} \|f\|^2_{L^1(K)} \|\sigma\|^2_{L^\infty(\mathbb{R})} V_m \lambda^m & \lambda \in [0,\vartheta) \\ \|f\|^2_{H^s(\Omega)} C^2_{\sigma,s}\left(-\frac{1}{2s}\lambda^{-2s} + \frac{2s+m}{2sm}\vartheta^{-2s}\right) & \lambda \in [\vartheta,\infty) \end{cases},
\end{aligned}$$

where the final bound is continuous at $\lambda = \vartheta$, and it is non-negative. □

D. Further Experiments

D.1. Simulations on a 2D artificial example

To further verify our results, we extend the 1D target function to a 2D case, which is expressed as follows:

$$f_{2D}(x_1,x_2;\sigma) = 0.2\exp\left(-\frac{(x_1-0.4)^2+(x_2-0.4)^2}{\sigma^2}\right) + 0.5\exp\left(-\frac{(x_1-0.6)^2+(x_2-0.6)^2}{\sigma^2}\right),$$

where $x_1 \in [0,1]$, $x_2 \in [0,1]$,$\sigma > 0$ is a scalar index that can determine the complexity of f_{2D}, similar as the 1D case.

Similar to Simulation 2 (1D case) as detailed in Section 5, we create different forms of target function $f(x_1,x_2;\sigma)$ by choosing σ as one element of the set $\{0.01, 0.05, 0.1, 0.5\}$, and for each regression task we build random nets with λ taken as an element from the set $\{0.1, 0.5, 1, 5, 10, 50, 100, 200\}$, and fix the number of hidden nodes as $L = 10000$ for each case. We sample 10000 instances $\{(x_1^{(i)}, x_2^{(i)}), f_{2D}(x_1^{(i)}, x_2^{(i)})\}_{i=1}^{10000}$ which are meshgrid points on $[0,1]^2$ (both x_1 and x_2 are equally space points over [0,1]), then randomly and uniformly select 5000 training samples and 5000 test samples.

For each pair (λ,σ), we run independently 50 trials and calculate the relative training error for each trial. The following Table 2 shows the averaged training performance for the case of each pair (λ,σ).

It is clear that similar findings can be seen from Table 2, that is, consistent with the conclusion drwan from Table 1, there exists an appropriate range/distribution $\mathcal{D}^*$, but **NOT ANY** range/distribution, such that a neural network with random weights (NNRWs) assigned from $\mathcal{D}^*$ can be a universal approximator. Essentially, the $\mathcal{D}^*$ (e.g, $[-\lambda^*, \lambda^*]^2$) is highly dependent upon the complexity of the target function, as consistent with the theoretical and empirical results elaborated in (Li & Wang, 2017).

Table 2. Summary of mean relative training error for various choices of (λ, σ) for the 2D case.

λ	Averaged Relative Training Error E			
	$\sigma = 0.01$	$\sigma = 0.05$	$\sigma = 0.1$	$\sigma = 0.5$
$\lambda = 0.1$	0.0310	0.0225	0.0121	0.0062
$\lambda = 0.5$	0.0297	0.0214	0.0086	0.0041
$\lambda = 1$	0.0296	0.0210	0.0072	0.0016
$\lambda = 5$	0.0277	0.0032	0.0012	2.8661e-04
$\lambda = 10$	0.0192	0.0011	3.4762e-05	2.1093e-05
$\lambda = 50$	0.0010	6.3672e-05	6.1358e-05	5.3784e-05
$\lambda = 100$	1.2561e-04	4.2462e-05	5.1378e-05	5.3165e-05
$\lambda = 200$	1.1762e-04	3.2672e-05	2.6826e-05	2.3018e-05

D.2. Simulations on five real-world datasets

Also, we conduct another simulation study on five real-world datasets from KEEL-dataset repository for regression task (`https://sci2s.ugr.es/keel/`). The basic information of these datasets is summarized in Table 3. We choose randomly 75% samples as traning set while the left samples for testing set. Similar as the experiments conducted on 1D and 2D artificial examples presented before, we also consider different settings of λ for each dataset, and fix $L = 10000$ for the neural network with random weights. Then, we run independently 50 trials and calculate the relative training error for each trial. The following Table 4 shows the averaged training performance for the case of each pair (λ, σ).

Table 3. Summary of basic information of five real-world datasets

Dataset	Number of Samples	Input Dimension	Output Dimension
stock	950	9	1
laser	993	4	1
friedman	1200	5	1
abalone	4177	8	1
compactiv	8192	21	1

Table 4. Summary of mean relative training error for various choices of (λ, σ) for real-world datasets.

λ	Averaged Relative Training Error E				
	stock	laser	friedman	abalone	compactiv
$\lambda = 0.1$	0.0065	0.0131	0.0314	0.0654	0.0145
$\lambda = 0.5$	1.4295e-09	0.0111	0.0027	0.0468	0.0029
$\lambda = 1$	1.0003e-10	0.0103	1.1831e-09	0.0300	9.8862e-04
$\lambda = 5$	2.2323e-13	5.8883e-04	1.8764e-13	3.5419e-09	3.0985e-08
$\lambda = 10$	2.6994e-14	4.2153e-10	9.1391e-15	9.7869e-11	5.8270e-09
$\lambda = 50$	3.0819e-15	4.4011e-14	3.1680e-15	1.5955e-13	2.3818e-10
$\lambda = 100$	2.8372e-15	8.9380e-15	2.9847e-15	6.2920e-14	5.3666e-11
$\lambda = 200$	5.2748e-15	2.9870e-15	3.1563e-15	1.6981e-14	3.0773e-11

It clearly shows that there are a few cases (like $\lambda = 0.1, 0.5, 1$) when the training errors of the randomized neural networks cannot converge to zero (even when $L = 10000$). This finding is also consist with what we have obtained in the 1D and 2D artificial examples. All these findings validate our theoretical results that when hidden parameters are distributed in a bounded domain, the network may not achieve zero approximation error.

D.3. Quantitative demonstration for Figure 1

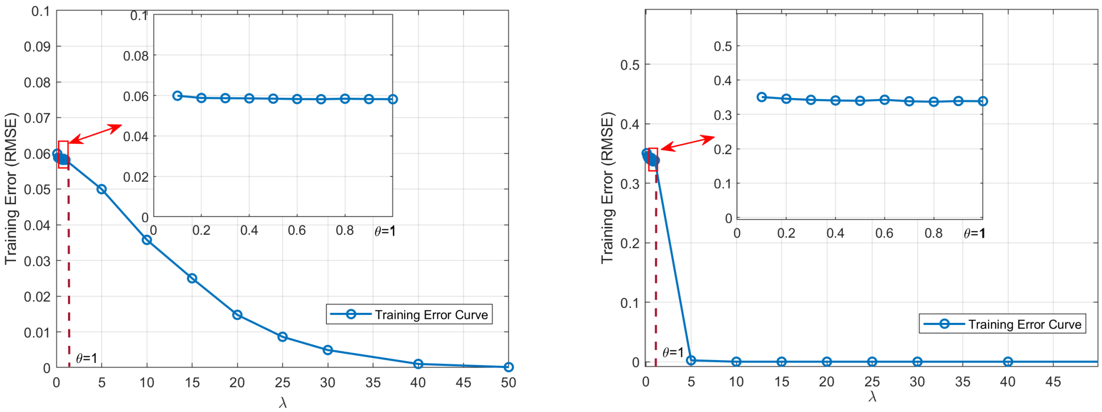
Consider the following two toy examples:

$$f_1(x) = 0.2e^{-(10x-4)^2} + 0.5e^{-(80x-40)^2} + 0.3e^{-(80x-20)^2},\ x \in [0, 1],$$

and

$$f_2(x) = 0.8\exp(-0.2x)\sin(10x),\ x \in [0,5].$$

We uniformly sample 1000 training samples (with $x \in [0,1]$ for f_1, $x \in [0,5]$ for f_2, respectively). For the neural networks with random weights (NNRWs), we fix the number of hidden nodes $L = 10000$ (so that we can observe the trend as $L \to \infty$) and try the randomized learner model using different setting of the random distribution, e.g., $\lambda = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 5, 10, 15, 20, 25, 30, 40, 50]$. As shown clearly in Figure 6 (similar to the qualitative plot shown in Figure 1), for both toy examples, when $\lambda = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1]$ the approximation error change is relatively flatten, while for $\lambda = [5, 10, 15, 20, 25, 30, 40, 50]$ the magnitude of the approximation error decreasing become much larger. Although it is intuitively seen that the threshold value ϑ is 'roughly' around 1 for both f_1 and f_2, it is not easy to find the 'optimal' value of ϑ. Given limited training samples (sampled from an unknown function), how to develop advanced algorithms/strategies to compute numerically the threshold ϑ is out of the focus of our current work. Nonetheless, it is expected to benefit and motivate future research on algorithm development for building more powerful (shallow and/or deep) neural nets with random weights (NNRWs).



(a) f_1

(b) f_2

Figure 6. Outline of the approximation lower bound for f_1 and f_2.