Effective Minkowski Dimension of Deep Nonparametric Regression: Function Approximation and Statistical Theories

Zixuan Zhang[1] **Minshuo Chen**[2] **Mengdi Wang**[2] **Wenjing Liao**[3] **Tuo Zhao**[1]

Abstract

Existing theories on deep nonparametric regression have shown that when the input data lie on a low-dimensional manifold, deep neural networks can adapt to the intrinsic data structures. In real world applications, such an assumption of data lying exactly on a low dimensional manifold is stringent. This paper introduces a relaxed assumption that the input data are concentrated around a subset of $\mathbb{R}^d$ denoted by $\mathcal{S}$, and the intrinsic dimension of $\mathcal{S}$ can be characterized by a new complexity notation – effective Minkowski dimension. We prove that, the sample complexity of deep nonparametric regression only depends on the effective Minkowski dimension of $\mathcal{S}$ denoted by p. We further illustrate our theoretical findings by considering nonparametric regression with an anisotropic Gaussian random design $N(0, \Sigma)$, where Σ is full rank. When the eigenvalues of Σ have an exponential or polynomial decay, the effective Minkowski dimension of such an Gaussian random design is $p = \mathcal{O}(\sqrt{\log n})$ or $p = \mathcal{O}(n^\gamma)$, respectively, where n is the sample size and $\gamma \in (0, 1)$ is a small constant depending on the polynomial decay rate. Our theory shows that, when the manifold assumption does not hold, deep neural networks can still adapt to the effective Minkowski dimension of the data, and circumvent the curse of the ambient dimensionality for moderate sample sizes.

1. Introduction

Deep learning has achieved impressive successes in various real-world applications, such as computer vision (Krizhevsky et al., 2012; Goodfellow et al., 2014; Long et al., 2015), natural language processing (Graves et al., 2013; Bahdanau et al., 2014; Young et al., 2018), and robotics (Gu et al., 2017). One notable example of this is in the field of image classification, where the winner of the 2017 ImageNet challenge achieved a top-5 error rate of just 2.25% (Hu et al., 2018) using a training dataset of 1 million labeled high resolution images in 1000 categories. Deep neural networks have been shown to outperform humans in speech recognition, with a 5.15% word error rate using the LibriSpeech training corpus (Panayotov et al., 2015), which consists of approximately 1000 hours of 16kHz read English speech from 8000 audio books.

The remarkable successes of deep learning have challenged conventional machine learning theory, particularly when it comes to high-dimensional data. Existing literature has established a minimax lower bound of sample complexity $n \gtrsim \epsilon^{-(2s+d)/s}$ for learning s-Hölder functions in $\mathbb{R}^d$ with accuracy ϵ (Györfi et al., 2006). This minimax lower bound, however, is far beyond the practical limits. For instance, the images in the ImageNet challenge are of the resolution $224 \times 224 = 50176$, while the sample size of 1.2 million is significantly smaller than the theoretical bound.

Several recent results have attempted to explain the successes of deep neural networks by taking the low-dimensional structures of data into consideration(Chen et al., 2019; 2022; Nakada & Imaizumi, 2020; Liu et al., 2021; Schmidt-Hieber, 2019). Specifically, Chen et al. (2022) shows that when the input data are supported on a p-dimensional Riemannian manifold embedded in $\mathbb{R}^d$, deep neural networks can capture the low-dimensional intrinsic structures of the manifold. The sample complexity in Chen et al. (2022) depends on the intrinsic dimension p, which circumvents the curse of ambient dimension d; Nakada & Imaizumi (2020) assumes that the input data are supported on a subset of $\mathbb{R}^d$ with Minkowski dimension p, and establishes a sample complexity similar to Chen et al. (2022). Liu et al. (2021) considers a classification problem, and show that convolutional residual networks enjoy similar

[1] School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA [2] Electrical and Computer Engineering, Princeton University, NJ, USA [3] School of Mathematics, Georgia Institute of Technology, Atlanta, GA, USA. Correspondence to: Tuo Zhao <tourzhao@gatech.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

theoretical properties to Chen et al. (2022).

Considering the complexity of real world applications, however, the assumptions of data lying exactly on a low-dimensional manifold or a set with low Minkowski dimension are stringent. To bridge such a gap between theory and practice, we consider a relaxed assumption that the input data X are approximately supported on a subset of $\mathbb{R}^d$ with certain low-dimensional structures denoted by $\mathcal{S}$. Roughly speaking, there exists a sufficiently small τ such that we have $\mathbb{P}(X \notin \mathcal{S}) = \tau$, where $\mathcal{S}$ can be characterized by a new complexity notation – effective Minkowski dimension. We then prove that under proper conditions, the sample complexity of nonparametric regression using deep neural networks only depends on the effective Minkowski dimension of $\mathcal{S}$ denoted by p. Our assumption arises from practical motivations: The distributions of real-world data sets often exhibit a varying density. In practice, the low-density region can be neglected, if our goal is to minimize the L_2 prediction error in expectation.

Furthermore, we illustrate our theoretical findings by considering nonparametric regression with an anisotropic multivariate Gaussian randomly sampled from $N(0, \Sigma)$ design in $\mathbb{R}^d$. Specifically, we prove that when the eigenvalues of Σ have an exponential decay, we can properly construct $\mathcal{S}$ with the effective Minkowski dimension $p = \min(\mathcal{O}(\sqrt{\log n}), d)$. Moreover, when the eigenvalues of Σ have a polynomial decay, we can properly construct $\mathcal{S}$ with the effective Minkowski dimension $p = \min(\mathcal{O}(n^\gamma, d))$, where $\gamma \in (0, 1)$ is a small constant. Our proposed effective Minkovski dimension is a non-trivial generalization of the manifold intrinsic dimension (Chen et al., 2022) or the Minkowski dimension (Nakada & Imaizumi, 2020), as both the intrinsic dimension or Minkowski dimension of the aforementioned $\mathcal{S}$'s are d, which can be significantly larger than p for moderate sample size n.

An ingredient in our analysis is an approximation theory of deep ReLU networks for β-Hölder functions (Yarotsky, 2017; Nakada & Imaizumi, 2020; Chen et al., 2019). Specifically, we show that, in order to uniformly approximate β-Hölder functions on a properly selected $\mathcal{S}$ up to an ϵ error, the network consists of at most $O(\epsilon^{-p/\beta})$ neurons and weight parameters, where p is the effective Minkowski dimension of the input data distribution. The network size in our theory only weakly depends on the ambient dimension d, which circumvents the curse of dimensionality for function approximation using deep ReLU networks. Our approximation theory is established for the L^2 norm instead of the L^∞ norm in Nakada & Imaizumi (2020); Chen et al. (2019). The benefit is that we only need to approximate the function accurately on the high-density region, and allow for rough approximations on the low-density region. Such flexibility is characterized by our effective Minkowski dimension.

The rest of this paper is organized as follows: Section 2 reviews the background; Section 3 presents our functional approximation and statistical theories; Section 4 provides an application to Gaussian random design; Section 5 presents the proof sketch of our main results; Section 6 discusses related works and draws a brief conclusion.

Notations Given a vector $v = (v_1, ..., v_d)^\top \in \mathbb{R}^d$, we define $\|v\|_p^p = \sum_j |v_j|^p$ for $p \in [1, \infty)$ and $\|v\|_\infty = \max_j |v_j|$. Given a matrix $W = [W_{ij}] \in \mathbb{R}^{n \times m}$, we define $\|W\|_\infty = \max_{i,j} |W_{ij}|$. We define the number of nonzero entries of v and W as $\|v\|_0$ and $\|W\|_0$, respectively. For a function $f(x)$, where $x \in \mathcal{X} \subseteq \mathbb{R}^d$, we define $\|f\|_\infty = \max_{x \in \mathcal{X}} |f(x)|$. We define $\|f\|_{L^2(P)}^2 = \int_{\mathcal{X}} f^2(x)p(x)dx$, where P is a continuous distribution defined on $\mathcal{X}$ with the pdf $p(x)$.

2. Background

In nonparametric regression, the aim is to estimate a ground-truth regression function f^* from i.i.d. noisy observations $\{(x_i, y_i)\}_{i=1}^n$. The data are generated via

$$y_i = f^*(x_i) + \xi_i,$$

where the noise ξ_i's are i.i.d. sub-Gaussian noises with $\mathbb{E}[\xi_i] = 0$ and variance proxy σ^2, which are independent of the x_i's. To estimate f^*, we minimize the empirical quadratic loss over a concept class $\mathcal{F}$, i.e.,

$$\widehat{f} \in \operatorname*{argmin}_{f \in \mathcal{F}} \frac{1}{2n} \sum_{i=1}^n (f(x_i) - y_i)^2. \tag{1}$$

We assess the quality of estimator $\widehat{f}$ through bounding L^2 distance between $\widehat{f}$ and f^*, that is,

$$\left\| \widehat{f} - f^* \right\|_{L^2(P_{\text{data}})}^2 \leq \gamma(n).$$

Here $\gamma(n)$ is a function of n describing the convergence speed and P_{data} is an unknown sampling distribution ofthe x_i's supported on $\mathcal{D}_{\text{data}}$.

Existing literature on nonparametric statistics has established an optimal rate of $\gamma(n) \lesssim n^{-\frac{2\alpha}{2\alpha+d}}$, when f^* is α-smooth with bounded functional norm, and $\mathcal{F}$ is properly chosen (Wahba, 1990; Altman, 1992; Fan & Gijbels, 1996; Tsybakov, 2008; Györfi et al., 2006).

The aforementioned rate of convergence holds for any data distribution P_{data}. For high-dimensional data, the convergence rate suffers from the curse of dimensionality. However, in many practical applications, P_{data} exhibits important patterns. For example, data are highly clustered in certain regions, while scarce in the rest of the domain. In literature, a line of work studies when P_{data} is supported on a low-dimensional manifold (Bickel & Li, 2007; Cheng & Wu, 2013; Liao et al., 2021; Kpotufe, 2011; Kpotufe & Garg, 2013; Yang et al., 2015). The statistical rate of convergence $\gamma(n)$ in these works depends on the intrinsic dimension of the manifold, instead of the ambient dimension. Recently, neural networks are also shown to be able to capture the low-dimensional structures of data (Schmidt-Hieber, 2019; Nakada & Imaizumi, 2020; Chen et al., 2022).

As mentioned, aforementioned works assume that data exactly lie on a low-dimensional set, which is stringent. Recently, Cloninger & Klock (2020) relaxes the assumption such that data are concentrated on a tube of the manifold, but the radius of this tube is limited to the reach (Federer, 1959) of the manifold. In this paper, we establish a fine-grained data dependent nonparametric regression theory, where data are approximately concentrated on a low-dimensional subset of the support.

To facilitate a formal description, we denote $\mathcal{D}_{\rm data}$ as the data support. Given $r, \tau > 0$, we define

$$N(r;\tau) := \inf_S \{N_r(S) : S \subset \mathcal{D}_{\rm data} \,\text{with}\, P_{\rm data}(S) \geq 1-\tau\},$$

where $N_r(S)$ is the r-covering number of S with respect to L^∞ distance.

Assumption 2.1. For any sufficiently small $r, \tau > 0$, there exists a positive constant $p = p(r, \tau)$ such that

$$\frac{\log N(r;\tau)}{-\log r} \leq p(r,\tau).$$

Furthermore, there exists $S \subset \mathcal{D}_{\rm data}$ such that

$$N_r(S) \leq c_0 N(r;\tau) \leq c_0 r^{-p}$$

for some constant $c_0 > 1$, $P_{\rm data}(S^c) \leq \tau$ and $|x_i| \leq R_S$ for any $x = (x_1, \ldots, x_d) \in S$ and some constant $R_S > 0$.

We next introduce Hölder functions and the Hölder space.

Definition 2.2 (Hölder Space)**.** Let $\beta > 0$ be a degree of smoothness. For $f : \mathcal{X} \to \mathbb{R}$, the *Hölder norm* is defined as

$$\|f\|_{\mathcal{H}(\beta,\mathcal{X})} := \max_{\alpha:\|\alpha\|_1 < \lfloor\beta\rfloor} \sup_{x\in\mathcal{X}} |\partial^\alpha f(x)| + \max_{\alpha:\|\alpha\|_1 = \lfloor\beta\rfloor} \sup_{x,x'\in\mathcal{X}, x\neq x'} \frac{|\partial^\alpha f(x) - \partial^\alpha f(x')|}{\|x-x'\|_\infty^{\beta-\lfloor\beta\rfloor}}.$$

Then the *Hölder space* on $\mathcal{X}$ is defined as

$$\mathcal{H}(\beta,\mathcal{X}) = \big\{f \in C^{\lfloor\beta\rfloor}(\mathcal{X}) \big| \, \|f\|_{\mathcal{H}(\beta,\mathcal{X})} \leq 1\big\}.$$

Without loss of generality, we impose the following assumption on the target function f^*:

Assumption 2.3. The ground truth function $f^* : \mathcal{D}_{\rm data} \to \mathbb{R}$ belongs to the Hölder space $\mathcal{H}(\beta, \mathcal{D}_{\rm data})$ with $\beta \in (0, d)$. Although the Hölder norm of f^* is assumed to be bounded by 1, our results can be easily extended to the case when $\|f^*\|_{\mathcal{H}(\beta,\mathcal{D}_{\rm data})}$ is upper bounded by any positive constant. In addition, $\beta < d$ is a natural assumption. Given that ambient dimension d is always large, it is unusual for regression functions to possess a degree of smoothness larger than d.

Our goal is to use multi-layer ReLU neural networks to estimate the function f^*. Given an input x, an L-layer ReLU neural network computes the output as

$$f(x) = W_L \cdot \mathrm{ReLU}(W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}) + b_L, \tag{2}$$

where $W_1, \ldots, W_L$ and $b_1, \ldots, b_L$ are weight matrices and intercepts respectively. The $\mathrm{ReLU}(\cdot)$ activation function denotes the entrywise rectified linear unit, i.e. $\mathrm{ReLU}(a) = \max\{a, 0\}$. The empirical risk minimization in (1) is taken over the function class $\mathcal{F}$ given by a network architecture.

Definition 2.4 (Function Class Given by a Network Architecture)**.** Given a tuple (L, B, K), a functional class of ReLU neural networks is defined as follows:

$$\mathcal{F}(L,B,K) := \big\{f | f(x) \text{ in the form of (2) with } L \text{ layers}, \|f\|_\infty \leq 1, \|W_i\|_\infty \leq B, \ \|b_i\|_\infty \leq B \text{ for } i = 1, \ldots, L, \sum_{i=1}^L \|W_i\|_0 + \|b_i\|_0 \leq K\big\}.$$

3. Approximation and Generalization Theory

In this section, we present generic approximation and generalization theory and defer detailed proofs to Section 5.1 and 5.2 respectively. Firstly, we introduce the approximation theory of utilizing deep neural networks to approximate Hölder functions. The approximation error is determined by effective Minkowski dimension of data distribution and probability of low-density area. Furthermore, we present the generalization error when approximating regression function f^*. The convergence rate also depends on effective Minkowski dimension.

Theorem 3.1 (Approximation of deep neural networks)**.** *Suppose Assumption 2.1 hold. For $\beta > 0$ and any sufficiently small $\epsilon, \tau > 0$, consider a tuple (L, B, K)*

$$L = C_1, \ B = O(R_S^{\beta s}\epsilon^{-s}), \text{ and } K = C_2(R_S d)^p \epsilon^{-p/\beta},$$

where $R_S > 0$ and $p = p(d^{-1}\epsilon^{1/\beta}/2, \tau)$ are given by Assumption 2.1, and

$$C_1 = O(d), \ C_2 = O\big(d^{2+\lfloor\beta\rfloor}\big), \text{ and } s = s(\beta).$$

Then for any $f^* \in \mathcal{H}(\beta, \mathcal{D}_{\rm data})$, we have

$$\inf_{f\in\mathcal{F}(L,B,K)} \|f - f^*\|^2_{L^2(P_{\rm data})} \leq \epsilon^2 + 4\tau.$$

The novelty of Theorem 3.1 is summarized below:

Dependence on Effective Minkowski Dimension. The approximation rate in Theorem 3.1 is $O(K^{-\beta/p})$, which only depends on effective Minkowski dimension $p < d$ and function smoothness β, but not on ambient dimension d. Compared to Yarotsky (2017), our results improves the exponential dependence of neural network size on d to that on p. Moreover, unlike Nakada & Imaizumi (2020) and Chen et al. (2022), our results do not require that data distribution is exactly supported on a low-dimensional structure. Instead, our results can work for data distribution with high-dimensional support as long as its effective Minkowski dimension is relatively small.

Relaxation to the L^2-error. The approximation error in Theorem 3.1 is established with respect to the $L^2(P_{\text{data}})$ norm, while most of existing works focus on the L^∞ error (Yarotsky, 2017; Nakada & Imaizumi, 2020; Chen et al., 2019). Intuitively, it is not necessary for the network class to approximate the function value at each point in the domain $\mathcal{D}_{\text{data}}$ precisely when data distribution is highly concentrated at certain subset. Instead, it suffices to approximate f^* where the probability density is significant, while the error for the low-density region can be easily controlled since the regression function f^* and the neural network class $f \in \mathcal{F}(L, B, K)$ are bounded.

The benefit of using the L^2 error is that, we only need to control the approximation error of f^* within some chosen region $S \subseteq \mathcal{D}_{\text{data}}$. Here S has an effective Minkowski dimension p, which ensures that it can be covered by $O(r^{-p})$ hypercubes with side length r. Then we design deep neural networks to approximate f^* within each hypercube and thus the network size depends on the number of hypercubes used to cover S. This explains why network size in Theorem 3.1 depends on p. Meanwhile, the probability out of S is negligible since the data density is low. We further demonstrate that this probability τ is far less than the approximation error in Section 4. By this means, we succeed to reduce the network size and at the same time achieve a small L^2 approximation error.

We next establish the generalization result for the estimation of f^* using deep neural networks.

Theorem 3.2 (Generalization error of deep neural networks)**.** *Suppose Assumption 2.1 holds. Fix any sufficiently small $r, \tau > 0$ satisfying $r < R_S$ and $\tau < r^{4\beta}/4$. Set a tuple (L, B, K) with C_1, C_2 and s appearing in Theorem 3.1 as*

$$L = C_1,\ B = O(R_S^{\beta s} r^{-\beta s}),\ \textit{and}\ K = C_2 R_S^p r^{-p}$$

with $p = p(r, \tau)$. Let $\widehat{f}$ be the global minimizer of empirical loss given in (1) with the function class $\mathcal{F} = \mathcal{F}(L, B, K)$. Then we have

$$\mathbb{E}\|\widehat{f} - f^*\|^2_{L^2(P_{\text{data}})} =$$
$$O\left(\tau + \sigma r^{2\beta} + \frac{\sigma^2}{n}\left(\frac{R_S}{r}\right)^p \log\left(\frac{(R_S/r)^p}{r^{4\beta} - 4\tau}\right)\right),$$

where $O(\cdot)$ hides polynomial dependence on d.

Theorem 3.2 is a statistical estimation result. It implies that the generalization error also depends on effective Minkowski dimension p. To establish this result, we decompose the squared error into a squared bias term and a variance term. The bias is tackled with the approximation error in Theorem 3.1 and the variance depends on the network size. With the network size growing, the variance term increases while the bias term decreases, since the approximation capability of neural networks is enhanced as the size of the network enlarges. Therefore, we need to trade off between the squared bias and the variance to minimize the squared generalization error.

Notably, our analysis in Section 3 holds for any sufficiently small τ and r, and every pair of τ and r determines a p. As shown in Assumption 2.1, if τ and r decreases, the covering number will become larger while the approximation can be more accurate. In order to establish an explicit bound, we need to trade off τ and r for the given sample size n. Therefore the "optimal" p eventually becomes functions of n. We call such an "optimal" p effective Minkowski dimension.

In the next section, we give two specific classes of Gaussian random designs to illustrate how effective Minkowski dimension $p(r, \tau)$ scales with r and τ. We further show that, under a proper selection of the region S and the covering accuracy r, the convergence rate for the estimation of f^* using deep neural networks is $\widetilde{O}(n^{-2\beta/(2\beta+p)})$, where the effective Minkowski dimension p is properly chosen.

4. Application to Gaussian Random Design

In literature, it is common to consider random Gaussian design in nonparametric regression (Anderson, 1962; Muller & Stewart, 2006; Chatfield, 2018). In this section, we take anisotropic multivariate Gaussian design as example to justify Assumption 2.1 and demonstrate the effective Minkowski dimension. Here we only provide our main theorems and lemmas. The detailed proofs are given in Section 5.3.

Consider a Gaussian distribution $P_{\text{data}} \sim N(\mathbf{0}, \Sigma)$ in $\mathbb{R}^d$. The covariance matrix Σ has the eigendecomposition form: $\Sigma = Q\Gamma Q^\top$, where Q is an orthogonal matrix and $\Gamma = \text{diag}(\gamma_1, \ldots, \gamma_d)$. For notational convenience in our analysis, we further denote eigenvalue $\gamma_i = \lambda_i^2$ for $i = 1, \ldots, d$. Without loss of generality, assume that

$\lambda_1^2 \geq \lambda_2^2 \geq \ldots \geq \lambda_d^2$. Furthermore, we assume that $\{\lambda_i^2\}_{i=1}^d$ has an exponential or polynomial decay rate:

Assumption 4.1 (Exponential decay rate)**.** The eigenvalue series $\{\gamma_i\}_{i=1}^d = \{\lambda_i^2\}_{i=1}^d$ satisfies $\lambda_i \leq \mu \exp\{-\theta i\}$ for some constants $\mu, \theta > 0$.

Assumption 4.2 (Polynomial decay rate)**.** The eigenvalue series $\{\gamma_i\}_{i=1}^d = \{\lambda_i^2\}_{i=1}^d$ satisfies $\lambda_i \leq \rho i^{-\omega}$ for some constants $\rho > 0$ and $\omega > 1$.

When the eigenvalues decay fast, the support of the data distribution $P_{\rm data}$ has degeneracy in some directions. In this case, the majority of probability lies in some region $S \subset \mathbb{R}^d$, which has an effective Minkowski dimension $p < d$. Specifically, consider a "thick" low-dimensional hyper-ellipsoid in $\mathbb{R}^d$,

$$S(R,r;p) := \left\{ Qz \,\middle|\, z = (z_1, \ldots, z_d) \in \mathbb{R}^d, \sum_{i=1}^{p} \frac{z_i^2}{\lambda_i^2} \leq R^2, \right.$$
$$\left. |z_j| \leq \frac{r}{2} \text{ for } j = p+1, \ldots, d \right\}, \quad (3)$$

where $R, r > 0$ and $p \in \mathbb{N}_+$ are independent parameters. For the simplicity of notation, we first define a standard hyper-ellipsoid and then linearly transform it to align with the distribution $N(0, \Sigma)$. The set $S(R,r;p)$ can be regarded as a hyper-ellipsoid scaled by $R > 0$ in the first p dimensions, and with thickness $r > 0$ in the rest $d-p$ dimensions. Then we construct a minimal cover as a union of nonoverlapping hypercubes with side length r for $S(R,r;p)$. The following lemma characterizes the relationship between the probability measure outside $S(R,r;p)$ and its covering number.

Lemma 4.3. *Given the eigenvalue series $\{\lambda_i^2\}_{i=1}^d$, for any $R, r > 0$, choose $p > 0$ such that $\lambda_p^{-1} = 2R/r$. If $p < R^2$, we will have*

$$\mathbb{P}(X \notin S(R,r;p)) = O\big(\exp(-R^2/3)\big),$$
$$N_r(S(R,r;p)) \leq \left(\frac{2R}{r}\right)^p \cdot \prod_{i=1}^{p} \lambda_i = \prod_{i=1}^{p} \left(\frac{\lambda_i}{\lambda_p}\right).$$

Remark 4.4. Since data distribution $P_{\rm data}$ is supported on $\mathbb{R}^d$, both the intrinsic dimension and the Minkowski dimension of $P_{\rm data}$ are d. However, Lemma 4.3 indicates that the effective Minkowski dimension of $P_{\rm data}$ is at most p.

According to Lemma 4.3, if we choose scale $R > \sqrt{p}$ properly, the probability outside S can be sufficiently small while the covering number of S is dominated by r^{-p}, which gives that the effective Minkowski dimension of $P_{\rm data}$ is at most p. Moreover, under fast eigenvalue decays, the product of the first p eigenvalues appearing in $N_r(S(R,r;p))$ is a small number dependent of p. In these cases, we specify the selection of R, r and p accordingly and show the effective Minkowski dimension is reduced to $p/2$ in Appendix D.

Furthermore, we remark that the effective Minkowski dimension p is not a fixed number given data distribution $P_{\rm data}$, but an increasing function of sample size n. As sample size n increases, the estimation accuracy of f^* is required to be higher, so that we are supposed to design more and smaller hypercubes to enable preciser estimation by neural networks. Besides, some of the $d-p$ dimensions are not negligible anymore and thereby become effective compared to the accuracy. Therefore, we need to incorporate more dimensions to be effective to achieve higher accuracy.

With this observation, we construct $S(R,r;p)$ such that its effective Minkowski dimension $p(n)$ increases while thickness $r(n)$ decreases as sample size n grows to enable preciser estimation. Then we develop the following sample complexity:

Theorem 4.5 (Generalization error under fast eigenvalue decay)**.** *Under Assumption 2.3, let $\widehat{f}$ be the global minimizer of empirical loss given in (1) with function class $\mathcal{F} = \mathcal{F}(L,B,K)$. Suppose Assumption 4.1 hold. Set a tuple (L,B,K) with the constants C_1, C_2 and s appearing in Theorem 3.1 as*

$$L = C_1, \quad B = O\left(n^{\frac{\beta s}{2\beta + \sqrt{\log n/\theta}}} (\log n)^{\beta s}\right),$$
$$\textit{and} \;\; K = C_2 n^{\frac{\sqrt{\log n/\theta}}{2\beta + \sqrt{\log n/\theta}}}.$$

Then we have

$$\mathbb{E}\left\|\widehat{f} - f^*\right\|_{L^2(P_{\rm data})}^2 = O\left(\sigma^2 n^{-\frac{2\beta(1-\eta)}{2\beta+\sqrt{\log n/\theta}}} (\log n)^{3/2}\right)$$

for sufficiently large n satisfying $\log(\log n)/\sqrt{\theta \log n} \leq \eta$, where $\eta > 0$ is an arbitrarily small constant. Moreover, suppose Assumption 4.2 hold instead. Set a tuple (L,B,K) as

$$L = C_1, \quad B = O\left(n^{\frac{(1+1/\omega)\beta s}{2\beta + n^{\kappa}}}\right),$$
$$\textit{and} \;\; K = C_2\left(n^{\frac{(1+1/\omega)n^{\kappa}/(2\beta+n^{\kappa})}{4\beta + 2n^{\kappa}}}\right),$$

where $\kappa = (1+1/\omega)/\omega$. Then we have

$$\mathbb{E}\left\|\widehat{f} - f^*\right\|_{L^2(P_{\rm data})}^2 = O\left(\sigma^2 n^{-\frac{2\beta}{2\beta + n^{(1+1/\omega)/\omega}}} \log n\right).$$

Theorem 4.5 suggests the effective Minkowski dimension of Gaussian distribution is $\sqrt{\log n/\theta}$ under exponential eigenvalue decay with speed θ and effective Minkowski dimension is $n^{(1+1/\omega)/\omega}$ under polynomial eigenvalue decay with speed ω. For moderate sample size n, i.e. effective Minkowski dimension is less that ambient dimension d, Theorem 4.5 achieves a faster convergence rate. When we have a vast amount of data, the effective Minkowski dimension is the same as the ambient dimension d, and then we can apply standard analysis of deep neural networks for d-dimensional inputs to obtain the convergence rate $\widetilde{O}(n^{-2\beta/(2\beta+d)})$. To the best of our knowledge, Theorem 4.5 appears to be the first result for nonparametric regression and deep learning theory, where the effective dimension varies with the sample size.

5. Proof Sketch

This section contains proof sketches of Theorem 3.1, 3.2 and Lemma 4.3.

5.1. Proof Sketch of Theorem 3.1

We provide a proof sketch of Theorem 3.1 in this part and defer technical details of the proof to Appendix A. The ReLU neural network in Theorem 3.1 is constructed in the following 5 steps:

1. Choose region $S \subset \mathcal{D}_{\text{data}}$ only on which we use ReLU neural networks to approximate f^*.
2. Construct a covering of S with hypercubes and then divide these hypercubes into several groups, so that neural networks constructed with respect to each group have nonoverlapping supports.
3. Implement ReLU neural networks to assign given input and estimated function value to corresponding hypercube.
4. Approximate f^* by a Taylor polynomial and then implement a ReLU neural network to approximate Taylor polynomial on each hypercube.
5. Sum up all the sub-neural-networks and take maximum to approximate f^*.

Step 1. Space separation. Firstly, we divide $\mathcal{D}_{\text{data}}$ into some region $S \subset \mathcal{D}_{\text{data}}$ with high probability measure and $S^c = \mathcal{D}_{\text{data}}$ with large volume. By Assumption 2.1, for any sufficiently small $r, \tau > 0$ and some constant $c_0 > 1$, there exists $S \subset \mathcal{D}_{\text{data}}$ such that $N_r(S) \leq c_0 N(r;\tau) \leq c_0 r^{-p}$ for some positive constant $p = p(r,\tau)$ and $P_{\text{data}}(S^c) \leq \tau$. Intuitively, we only need to approximate f^* on S while S^c is negligible due to its small probability measure. Therefore, in the following steps, we only design a covering for S and approximate f^* in each hypercube of the covering.

Step 2. Grouping hypercubes. Let $\mathcal{C}$ be a minimum set of hypercubes with side length r covering S. Then we partition $\mathcal{C}$ into $\mathcal{C}_1, \dots, \mathcal{C}_J$ such that each subset $\mathcal{C}_j$ is composed of hypercubes separated by r from each other. Lemma A.1 shows that the number of $\mathcal{C}_j$'s is at most a constant dependent of d.

As a consequence, we group hypercubes into several subsets of $\mathcal{C}$ so that constructed neural networks with respect to each hypercube in $\mathcal{C}_j$ have nonoverlapping support.

Step 3. Hypercube Determination. This step is to assign the given input x and estimated function value y to the hypercube where they belong. To do so, we design a neural network to approximate function $(x,y) \mapsto y\,\mathbb{1}_I(x)$ where $I \in \mathcal{C}$ is some hypercube. To make functions positive, we firstly consider approximating $f_0 = f^* + 2$. Notice that $f_0 \in \mathcal{H}(\beta, \mathcal{D}_{\text{data}}, 3)$ and $1 \leq f_0(x) \leq 3$ for any $x \in \mathcal{D}_{\text{data}}$.

For any fixed $I \in \mathcal{C}$, we define the center of I as $(\iota_1, \dots, \iota_d)$. Then we construct a neural network $g_I^{\text{ind},r} : \mathcal{D}_{\text{data}} \times \mathbb{R}_{\geq} \to \mathbb{R}_{\geq}$ with the form:

$$g_I^{\text{ind},r}(x,y) = 4\,\text{ReLU}\left(\sum_{i=1}^{d} \widehat{\mathbb{1}}_{I,i}^{r}(x_i) + \frac{y}{4} - d\right), \tag{4}$$

where $\widehat{\mathbb{1}}_{I,i}^{r} : \mathbb{R} \to [0,1]$ is the approximated indicator function given by

$$\widehat{\mathbb{1}}_{I,i}^{r}(z) = \begin{cases} \frac{z-(\iota_i - r)}{r/2} & \text{if } \iota_i - r < z \leq \iota_i - \frac{r}{2}, \\ 1 & \text{if } \iota_i - \frac{r}{2} < z \leq \iota_i + \frac{r}{2}, \\ \frac{(\iota_i + r) - z}{r/2} & \text{if } \iota_i + \frac{r}{2} < z \leq \iota_i + r, \\ 0 & \text{otherwise.} \end{cases}$$

We claim that neural network $g_I^{\text{ind},r}$ approximates function $(x,y) \mapsto y\,\mathbb{1}_I(x)$. Moreover, Appendix A.2 provides the explicit realization of $g_I^{\text{ind},r}$ by selecting specific weight matrices and intercepts.

Step 4. Taylor Approximation. In each cube $I \in \mathcal{C}$, we locally approximate f^* by a Taylor polynomial of degree $\lfloor \beta \rfloor$ and then we define a neural network to approximate this Taylor polynomial. Firstly, we cite the following lemma to evaluate the difference between any β-Hölder function and its Taylor polynomial:

Lemma 5.1 (Lemma A.8 in Petersen & Voigtlaender (2018))**.** *Fix any $f \in \mathcal{H}(\beta, \mathcal{D}_{\text{data}})$ with $\|f\|_{\mathcal{H}(\beta,\mathcal{D}_{\text{data}})} \leq 1$ and $\bar{x} \in S$. Let $\bar{f}(x)$ be the Taylor polynomial of degree $\lfloor \beta \rfloor$ of f around $\bar{x}$, namely,*

$$\bar{f}(x) = \sum_{|\alpha| \leq \lfloor \beta \rfloor} \frac{\partial^\alpha f(\bar{x})}{\alpha!}(x - \bar{x})^\alpha.$$

Then, $|f(x) - \bar{f}(x)| \leq d^\beta \|x - \bar{x}\|^\beta$ holds for any $x \in \mathcal{D}_{\text{data}}$.

Next, we design an m-dimensional multiple output neural network $g_\epsilon^{\text{poly}} = (g_{\epsilon,1}^{\text{poly}}, \dots, g_{\epsilon,m}^{\text{poly}})$ to estimate multiple Taylor polynomials in each output. The existence of such neural network is ensured in the following lemma, which is a straightforward extension of Lemma 18 in Nakada & Imaizumi (2020).

Lemma 5.2 (Taylor approximation on S)**.** *Fix any $m \in \mathbb{N}_+$. Let $\{c_{k,\alpha}\} \subset [-1,1]$ for $1 \leq k \leq m$. Let $\{x_k\}_{k=1}^m \subset S$. Then there exist $c_1^{\text{poly}} = c_1^{\text{poly}}(\beta, d, p)$,*

$c_2^{\text{poly}} = c_2^{\text{poly}}(\beta, d, p)$ and $s_1^{\text{poly}} = s_1^{\text{poly}}(\beta, d, p)$ such that for any sufficiently small $\epsilon > 0$, there is a neural network g_ϵ^{poly} which satisfies the followings:

1. *$\sup_{x\in S}\left|g_{\epsilon,k}^{\text{poly}}(x) - \sum_{|\alpha|<\beta} c_{k,\alpha}(x - x_k)^\alpha\right| \le \epsilon$ for any $k = 1, \ldots, m$,*
2. *$L(g_\epsilon^{\text{poly}}) \le 1 + (2 + \log_2 \beta)(11 + (1+\beta)/p)$,*
3. *$B(g_\epsilon^{\text{poly}}) \le c_1^{\text{poly}} R_S^{\beta s_1^{\text{poly}}} \epsilon^{-s_1^{\text{poly}}}$,*
4. *$K(g_\epsilon^{\text{poly}}) \le c_2^{\text{poly}}(R_S^p \epsilon^{-p/\beta} + m)$.*

For any cube $I_k \in \mathcal{C}$, we take $f_{I_k}(x)$ as a Taylor polynomial function as with setting $\bar{x} \leftarrow x_{I_k}$ and $f \leftarrow f_0$ in Lemma 5.1. Then we define a neural network to approximate f_{I_k}, which is an $\epsilon/2$-accuracy Taylor polynomial of f_0. Let $g_{\epsilon/2}^{\text{poly}}$ be a neural network constructed in Lemma 5.2 with $\epsilon \leftarrow \epsilon/2$, $m \leftarrow N_r(S)$, $(x_k)_{k=1}^m \leftarrow (x_{I_k})_{k=1}^{N_r(S)}$, and $(c_{k,\alpha})_{k=1}^m \leftarrow (\partial^\alpha f(x_{I_k})/\alpha!)_{k=1}^{N_r(S)}$ appearing in Lemma 5.2. Then, we obtain

$$\sup_{k=1,\ldots,N_r(S)} \sup_{x\in S}\left|f_{I_k}(x) - g_{\epsilon/2,k}^{\text{poly}}(x)\right| \le \frac{\epsilon}{2}. \tag{5}$$

In addition, we construct a neural network to aggregate the outputs of $g_{\epsilon/2}^{\text{poly}}$. Define a neural network $g_k^{\text{filter}} : \mathbb{R}^{d+N_r(S)} \to \mathbb{R}^{d+1}$ which picks up the first d inputs and $(d+k)$-th input as

$$g_k^{\text{filter}}(z) = \begin{pmatrix} \mathbb{I}_d & e_k^\top \\ \mathbf{0_d} & e_k^\top \end{pmatrix} z, \;\; \text{for } k = 1, \ldots, N_r(S).$$

Then we design a neural network $g_{\epsilon/2}^{\text{simul}} : \mathbb{R}^d \to \mathbb{R}^{N_r(S)}$ that simultaneously estimates Taylor polynomial at each cube. Specifically, $g_{\epsilon/2}^{\text{simul}}$ is formulated as below

$$\begin{aligned} g_{\epsilon/2}^{\text{simul}} =& \big(g_{I_1}^{\text{ind},r} \circ g_1^{\text{filter}}, \ldots, g_{I_{N_r(S)}}^{\text{ind},r} \circ g_{N_r(S)}^{\text{filter}}\big) \\ & \circ \big(g_{d,L}^{Id}, g_{\epsilon/2}^{\text{poly}}\big), \end{aligned} \tag{6}$$

where $g_{d,L}^{Id} : \mathbb{R}^d \to \mathbb{R}^d$ is the neural network version of the identity function whose number of layers is equal to $L(g_{\epsilon/2}^{\text{poly}})$.

Step 5. Construction of Neural Networks. In this step, we construct a neural network $g_\epsilon^{f_0}$ to approximate $f_0 = f^* + 2$. Let $g^{\max,5^d}$ be the neural network version of the maximize function over 5^d numbers. Besides, define

$$g^{\text{sum}}(z_1, \ldots, z_{N_r(S)}) = \left(\sum_{I_k\in\mathcal{C}_1} z_k, \ldots, \sum_{I_k\in\mathcal{C}_{N_r(S)}} z_k\right),$$

which aims to sum up the output of $g_{\epsilon/2}^{\text{simul}}$ in each subset of covering $\mathcal{C}_j$.

Now we are ready to define $g_\epsilon^{f_0}$. Let $g_\epsilon^{f_0} := g^{\max,5^d} \circ g^{\text{sum}} \circ g_{\epsilon/2}^{\text{simul}}$. Equivalently, $g_\epsilon^{f_0}$ can be written as $g_\epsilon^{f_0} = \max_{j\in[5^d]} \sum_{I_k\in\mathcal{C}_j} g_{\epsilon/2,k}^{\text{simul}}$. Then we come to bound the approximation error of $g_\epsilon^{f_0}$. When $x \in S$, there exists some $I \in \mathcal{C}$ such that $x \in I$. Based on the method to construct neural networks, we have

$$\begin{aligned} g_\epsilon^{f_0}(x) &= \max_{I_k\in\text{Neig}(I)} g_{\epsilon/2,k}^{\text{simul}}(x) \\ &\le \max_{I_k\in\text{Neig}(I)} g_{\epsilon/2,k}^{\text{poly}}(x), \end{aligned}$$

where $\text{Neig}(I) = \{I' \in \mathcal{C} | (I \oplus 3r/2) \cap I' \neq \emptyset\}$ denotes the $3r/2$-neighborhood of hypercube I. In other words, when computing $g_\epsilon^{f_0}(x)$, we only need to take maximum over the estimated function value within hypercubes near x.

Given sufficiently small $\epsilon > 0$, the error is bounded as

$$\begin{aligned} |g_\epsilon^{f_0}(x) - f_0(x)| \le& \max_{I_k\in\text{Neig}(I)} \left|g_{\epsilon/2,k}^{\text{poly}}(x) - f_0(x)\right| \\ \le& \max_{I_k\in\text{Neig}(I)} \left|g_{\epsilon/2,k}^{\text{poly}}(x) - f_{I_k}(x)\right| \\ &+ \max_{I_k\in\text{Neig}(I)} \left|f_{I_k}(x) - f_0(x)\right| \\ \le& \frac{\varepsilon}{2} + d^\beta \left(\frac{3r}{2}\right)^\beta \le \epsilon, \end{aligned}$$

where the last inequality follows from (5) and Lemma 5.1. Detailed derivation of approximation error is deferred to Appendix A.3. In terms of parameter tuning, we choose $r = d^{-1}\epsilon^{1/\beta}/2$.

To extend results of f_0 to f^*, we implement a neural network $g^{\text{mod}}(z) = (-z+1) \circ \text{ReLU}(-z+2) \circ \text{ReLU}(z-1)$ and consider $g_\epsilon^{f^*} = g^{\text{mod}} \circ g_\epsilon^{f_0}$ to obtain the desired approximation error ϵ for any $x \in S$. Then we evaluate the approximation error with respect to L^2-norm:

$$\begin{aligned} &\left\|g_\epsilon^{f^*} - f^*\right\|_{L^2(P_{\text{data}})} \\ &= \left(\int_S + \int_{S^c}\right)\left(g_\epsilon^{f^*}(x) - f^*(x)\right)^2 \mathrm{d}P_{\text{data}}(x) \\ &\le \epsilon^2 + 4\tau. \end{aligned}$$

This follows from the aforementioned approximation error within S, boundedness of f^* and neural networks, as well as the property that out-of-S probability is upper bounded, i.e. $P_{\text{data}}(S^c) \le \tau$.

Finally, we sum up sizes of all the sub-neural-networks and thus obtain the network size of $g_\epsilon^{f^*}$. See Appendix A.4 for detailed calculation.

5.2. Proof Sketch of Theorem 3.2

Proof of Theorem 3.2 follows a standard statistical decomposition, i.e. decomposing the mean squared error of estimator $\widehat{f}$ into a squared bias term and a variance term. We bound the bias and variance separately, where the bias is tackled by the approximation results given in Theorem 3.1 and the variance is bounded using the metric entropy arguments. Details of the proof for Theorem3.2 are provided in Appendix B. At first, we decompose the L_2 risk as follows:

$$\mathbb{E}\left\|\widehat{f}-f^*\right\|^2_{L^2(P_{\text{data}})} = \underbrace{2\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(\widehat{f}(x_i)-f^*(x_i))^2\right]}_{T_1}$$
$$\underbrace{+\mathbb{E}\left\|\widehat{f}-f^*\right\|^2_{L^2(P_{\text{data}})} - 2\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(\widehat{f}(x_i)-f^*(x_i))^2\right]}_{T_2},$$

where T_1 reflects the squared bias of using neural networks to estimate f^* and T_2 is the variance term.

Step 1. Bounding bias term $\mathbf{T_1}$. Since T_1 is the empirical L_2 risk of $\widehat{f}$ evaluated on the samples $\{x_i\}_{i=1}^n$, we relate T_1 to the empirical risk by rewriting $f^*(x_i) = y_i - \xi_i$, so that we can apply the approximation error to bound the minimal empirical risk achieved by $\widehat{f}$. After some basic calculation, we have

$$T_1 \leq 2 \inf_{f\in\mathcal{F}(L,B,K)} \|f(x)-f^*(x)\|^2_{L^2(P_{\text{data}})} + 4\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\xi_i\widehat{f}(x_i)\right].$$

Note that the first term is the squared approximation error of neural networks, which can be controlled by Theorem 3.1. We bound the second term by quantifying the complexity of the network class $\mathcal{F}(L,B,K)$. A precise upper bound of T_1 is given in the following lemma.

Lemma 5.3. *Fix the neural network class $\mathcal{F}(L,B,K)$. For any $\delta \in (0,1)$, there exists some constant $c > 0$, such that*

$$\begin{aligned} T_1 \leq& c \inf_{f\in\mathcal{F}(L,B,K)} \|f(x)-f^*(x)\|^2_{L^2(P_{\text{data}})} \\ &+ c\sigma^2\frac{\log\mathcal{N}_2(\delta,\mathcal{F}(L,B,K))+2}{n} \\ &+ c\left(\sqrt{\frac{\log\mathcal{N}_2(\delta,\mathcal{F}(L,B,K))+2}{n}}+1\right)\sigma\delta, \end{aligned}$$

where $\mathcal{N}_2(\delta,\mathcal{F}(L,B,K))$ denotes the δ-covering number of $\mathcal{F}(L,B,K)$ with respect to the L^2 norm, i.e., there exists a discretization of $\mathcal{F}(L,B,K)$ into $\mathcal{N}_2(\delta,\mathcal{F}(L,B,K))$ distinct elements, such that for any $f\in\mathcal{F}$, there is $\bar{f}$ in the discretization satisfying $\left\|\bar{f}-f\right\|_2 \leq \epsilon$.

Step 2. Bounding variance term $\mathbf{T_2}$. We observe that T_2 is the difference between the population risk of $\widehat{f}$ and its empirical risk. However, bounding this difference is distinct from traditional concentration results because of the scaling factor 2 before the empirical risk. To do this, we divide the empirical risk into two parts and use a higher-order moment (fourth moment) to bound one part. Using a Bernstein-type inequality, we are able to show that T_2 converges at a rate of $1/n$, and the upper bound for this is shown in the following lemma.

Lemma 5.4. *For any $\delta \in (0,1)$, there exists some constant $c' > 0$, such that*

$$T_2 \leq \frac{c'}{3n}\log\mathcal{N}_2(\delta/4H,\mathcal{F}(L,B,K)) + c'\delta.$$

Step 3. Covering number of neural networks. The upper bounds of T_1 and T_2 in Lemma 5.3 and 5.4 both rely on the covering number of the network class $\mathcal{F}(R,\kappa,L,p,K)$. In this step, we present an upper bound for the covering number $\mathcal{N}_2(\delta,\mathcal{F}(L,B,K))$ for a given a resolution $\delta > 0$.

Lemma 5.5 (Covering number bound for $\mathcal{F}$ (Lemma 21 in Nakada & Imaizumi (2020))). *Given $\delta > 0$, the δ-covering number of the neural network class $\mathcal{F}(L,B,K)$ satisfies*

$$\log\mathcal{N}_2(\delta,\mathcal{F}(L,B,K)) \leq K\log\left(\frac{2^L\sqrt{dL}K^{L/2}B^LR_S}{\sqrt{\delta^2-4\tau}}\right).$$

Step 4. Bias-Variance Trade-off. Now we are ready to finish the proof of Theorem 3.2. Combining the upper bounds of T_1 in Lemma 5.3 and T_2 in Lemma 5.4 together and substituting the covering number in Lemma 5.5, we obtain

$$\mathbb{E}\left\|\widehat{f}-f^*\right\|^2_{L^2(P_{\text{data}})} = O\left(\tau + d^{2\beta}r^{2\beta} + \sigma\delta + \frac{\sigma^2K}{n}\log\left(\frac{\sqrt{dL}K^{L/2}B^LR_S}{\sqrt{\delta^2-4\tau}}\right)\right),$$

where we set approximation error to be $d^{2\beta}r^{2\beta}$. Plug in our choice of (L,B,K), and choose $\delta = r^{2\beta}$. Then we can conclude

$$\mathbb{E}\left\|\widehat{f}-f^*\right\|^2_{L^2(P_{\text{data}})} = O\left(\tau + \sigma r^{2\beta} + \frac{\sigma^2}{n}\left(\frac{R_S}{r}\right)^p\log\left(\frac{(R_S/r)^p}{r^{4\beta}-4\tau}\right)\right).$$

5.3. Proof Sketch of Lemma 4.3

In this section, we present our basic idea to construct $S(R,r;p)$ and the proof sketch of Lemma 4.3. For simplicity of proof, we assume $Q = I$ so that $\Sigma = \Lambda = \text{diag}(\lambda_1^2,\ldots,\lambda_d^2)$. The detailed proof is given in Appendix C, which can be easily extended to the case when Q is not an identity matrix. The proof of Theorem 4.5 is given in Appendix D.

Given the Gaussian sample distribution, we hope to choose some region in $S \subset \mathbb{R}^d$ with high probability measure and effective Minkowski dimension $p < d$. Then we can only apply neural networks to approximate f^* within each cube of the small covering of S and thereby significantly reduce the network size. In literature, it is common to truncate the ambient space within a hyper-ellipsoid for Gaussian distribution (Ellis & Maitra, 2007; Pakman & Paninski, 2014). Similarly, we consider the 'thick' low-dimensional hyper-ellipsoid $S(R, r; p)$ defined in (3). Then we construct a minimal cover of $S(R, r; p)$ as a union of nonoverlapping hypercubes with side length r, which is equal to the thickness of $S(R, r; p)$. In particular, this cover contains multiple layers of hypercubes to cover the first p dimensions of $S(R, r; p)$ while only needs one layer for the rest dimensions. Intuitively, we only learn about hypercubes that cover the first p dimensions without paying extra effort to study dimensions with low probability density.

A natural question arising from the construction of $S(R, r; p)$ is how to select a proper dimension p. To address this problem, we first notice that each side length of the p-dimension hyper-ellipsoid is supposed to be greater than the side length of hypercubes r, i.e. $2\lambda_i R \geq r$ for $i = 1, \ldots, p$, so that we would not waste hypercubes to cover dimensions with too small side length. For simplicity of calculation, we choose $p > 0$ that satisfies $\lambda_p^{-1} = 2R/r$ for any given $R, r > 0$.

Now we come to prove Lemma 4.3. Firstly, we compute the probability outside $S(R, r; p)$. By union bound, this probability can be upper bounded by two parts, the probability out of hyper-ellipsoid for the first p dimensions and the probability out of hypercube with side length r for the rest $d - p$ dimensions. The first part is equal to the tail bound of p-dimensional standard Gaussian by the construction of $S(R, r; p)$. The second part can be solved similarly by linearly transforming each dimension to be standard Gaussian.

Then we calculate the covering number of $S(R, r; p)$. Notice that the first p dimensions of $S(R, r; p)$ is contained in a p-dimensional hyper-rectangle with side length $2\lambda_i R$ for $i = 1, \ldots, p$, while only one hypercube is required to cover the j-th dimension for $j = p + 1, \ldots, d$. Therefore, the r-covering number can be upper bounded by $\prod_{i=1}^{p}(2\lambda_i R)/r^p$.

6. Discussion and Conclusion

In this paper, we have presented a generic approximation and generalization theory and applied it to Gaussian Random Design. Furthermore, our theory is applicable to scenarios where data are sampled from a mixture of distributions, denoted as $P_{\text{data}} = \sum_i w_i P_i$. Each distribution P_i is assigned a weight w_i and has a low-dimensional support. In such cases, we focus on a subset of distributions with significant weights, while neglecting distributions with small weights. Then the effective Minkowski dimension depends on the data support of the selected distributions. As the sample size grows, including more distributions becomes necessary to achieve higher estimation accuracy. Another example where our theory can be applied is the case of an approximate manifold, where the data are concentrated on a low-dimensional manifold. In this scenario, the effective Minkowski dimension corresponds to the intrinsic dimension of the manifold.

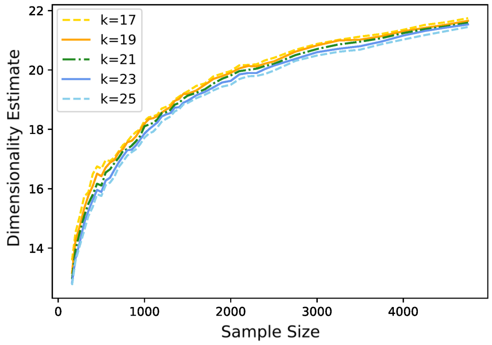

Figure 1. Dimensionality estimates of images obtained using the MLE method with k nearest neighbors under different sample size.

To illustrate the concept of effective dimension in real-world examples, we refer to a study by Pope et al. (2021), which investigates the intrinsic dimension of several popular benchmark datasets for deep learning. The intrinsic dimension presented in Pope et al. (2021) can be seen as an approximate estimate of the Minkowski dimension, as demonstrated in Levina & Bickel (2004); Grassberger & Procaccia (1983). In our work, we adopt the methodology employed by Pope et al. (2021) and utilize generative adversarial networks trained on the ImageNet dataset to generate samples containing varying numbers of daisy images. To estimate the intrinsic dimension of these generated samples, we employ the Maximum Likelihood Estimation (MLE) method, which is achieved by computing the Euclidean distances between each data point and its k nearest neighbors. The obtained results are presented in Figure 1, which clearly demonstrates that the intrinsic dimension estimated from a finite sample of images increases as the sample size grows. This finding aligns with our theory that the effective Minkowski dimension is an increasing function of the sample size.

In conclusion, this paper studies nonparametric regression of functions supported in $\mathbb{R}^d$ under data distribution with effective Minkowski dimension $p < d$, using deep neural networks. Our results show that the L^2 error for the estimation of $f^* \in \mathcal{H}(\beta, \mathcal{D}_{\text{data}})$ converges in the order of $n^{-2\beta/(2\beta+p)}$. To obtain an ϵ-error for the estimation of f^*, the sample complexity scales in the order of $\epsilon^{-(2\beta+p)/\beta}$, which demonstrates that deep neural networks can capture the effective Minkowski dimension p of data distribution. Such results can be viewed as theoretical justifications for the empirical success of deep learning in various real-world applications where data are approximately concentrated on a low-dimensional set.

References

Altman, N. S. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 1992.

Anderson, T. W. An introduction to multivariate statistical analysis. 1962.

Bahdanau, D., Cho, K., and Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Bickel, P. J. and Li, B. Local polynomial regression on unknown manifolds. *Lecture Notes-Monograph Series*, 54:177–186, 2007.

Chatfield, C. *Introduction to multivariate analysis*. Routledge, 2018.

Chen, M., Jiang, H., Liao, W., and Zhao, T. Efficient approximation of deep relu networks for functions on low dimensional manifolds. *Advances in neural information processing systems*, 32, 2019.

Chen, M., Jiang, H., Liao, W., and Zhao, T. Nonparametric regression on low-dimensional manifolds using deep relu networks: Function approximation and statistical recovery. *Information and Inference: A Journal of the IMA*, 11 (4):1203–1253, 2022.

Cheng, M.-Y. and Wu, H.-t. Local linear regression on manifolds and its geometric interpretation. *J. Amer. Statist. Assoc.*, 108(504):1421–1434, 2013.

Cloninger, A. and Klock, T. Relu nets adapt to intrinsic dimensionality beyond the target domain. *arXiv preprint arXiv:2008.02545*, 2020.

Ellis, N. and Maitra, R. Multivariate gaussian simulation outside arbitrary ellipsoids. *Journal of Computational and Graphical Statistics*, 16(3):692–708, 2007.

Fan, J. and Gijbels, I. *Local Polynomial Modelling and Its Applications*. Monographs on statistics and applied probability series. Chapman & Hall, 1996.

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

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Grassberger, P. and Procaccia, I. Measuring the strangeness of strange attractors. *Physica D: nonlinear phenomena*, 9(1-2):189–208, 1983.

Graves, A., Mohamed, A.-r., and Hinton, G. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pp. 6645–6649. IEEE, 2013.

Gu, S., Holly, E., Lillicrap, T., and Levine, S. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 3389–3396. IEEE, 2017.

Györfi, L., Kohler, M., Krzyzak, A., and Walk, H. *A distribution-free theory of nonparametric regression*. Springer Science & Business Media, 2006.

Hu, J., Shen, L., and Sun, G. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018.

Kpotufe, S. k-NN regression adapts to local intrinsic dimension. In Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 24*, pp. 729–737. 2011.

Kpotufe, S. and Garg, V. K. Adaptivity to local smoothness and dimension in kernel regression. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 26*, pp. 3075–3083. 2013.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Levina, E. and Bickel, P. Maximum likelihood estimation of intrinsic dimension. *Advances in neural information processing systems*, 17, 2004.

Liao, W., Maggioni, M., and Vigogna, S. Multiscale regression on unknown manifolds. *arXiv preprint arXiv:2101.05119*, 2021.

Liu, H., Chen, M., Zhao, T., and Liao, W. Besov function approximation and binary classification on low-dimensional manifolds using convolutional residual networks. In *International Conference on Machine Learning*, pp. 6770–6780. PMLR, 2021.

Long, J., Shelhamer, E., and Darrell, T. Fully convolutional networks for semantic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

Muller, K. E. and Stewart, P. W. *Linear model theory: univariate, multivariate, and mixed models*. John Wiley & Sons, 2006.

Nakada, R. and Imaizumi, M. Adaptive approximation and generalization of deep neural network with intrinsic dimensionality. *J. Mach. Learn. Res.*, 21(174):1–38, 2020.

Pakman, A. and Paninski, L. Exact hamiltonian monte carlo for truncated multivariate gaussians. *Journal of Computational and Graphical Statistics*, 23(2):518–542, 2014.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210. IEEE, 2015.

Petersen, P. and Voigtlaender, F. Optimal approximation of piecewise smooth functions using deep relu neural networks. *Neural Networks*, 108:296–330, 2018. ISSN 0893-6080.

Pope, P., Zhu, C., Abdelkader, A., Goldblum, M., and Goldstein, T. The intrinsic dimension of images and its impact on learning. *arXiv preprint arXiv:2104.08894*, 2021.

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

Tsybakov, A. B. *Introduction to Nonparametric Estimation*. Springer Publishing Company, Incorporated, 1st edition, 2008.

Wahba, G. *Spline models for observational data*, volume 59. Siam, 1990.

Yang, Y., Tokdar, S. T., et al. Minimax-optimal nonparametric regression in high dimensions. *Ann. Statist.*, 43(2): 652–674, 2015.

Yarotsky, D. Error bounds for approximations with deep relu networks. *Neural Networks*, 94:103–114, 2017. ISSN 0893-6080.

Young, T., Hazarika, D., Poria, S., and Cambria, E. Recent trends in deep learning based natural language processing. *ieee Computational intelligenCe magazine*, 13(3):55–75, 2018.

A. Proof of Theorem 3.1

In this section, we provide the omitted proof in Section 5.1.

A.1. Lemma A.1

Lemma A.1 (Lemma 20 in Nakada & Imaizumi (2020)). *Let $\mathcal{C} = \{I_k\}_{k=1}^{N_r(S)}$ be a minimum r-covering of S where I_k's are hypercubes with side length r. Then, there exists a disjoint partition $\{\mathcal{C}_j\}_{j=1}^{5^d} \subset \mathcal{C}$ such that $\mathcal{C} = \bigcup_{j=1}^{5^d} \mathcal{C}_j$ and $d(I_i, I_l) \geq r$ hold for any $I_i \neq I_l \in \mathcal{C}_j$ if $\mathrm{card}(\mathcal{C}_j) \geq 2$, where $d(\mathcal{A}, \mathcal{B}) := \inf\{\|x - y\| \,|x \in \mathcal{A}, y \in \mathcal{B}\}$ is defined as the distance of any two sets $\mathcal{A}$ and $\mathcal{B}$.*

A.2. Realization of hypercube determination function $g_I^{\text{ind},r}$

Hypercube determination function $g_I^{\text{ind},r}$ can be realized by weight matrices and intercepts $(4, 0) \odot (W^2, -d) \odot [(W_1^1, b_1^1), \ldots, (W_d^1, b_d^1)]$ where W_i^1, b_i^1 and W^2 are defined by

$$W_i^1 := \begin{pmatrix} e_i^\top & e_i^\top & e_i^\top & e_i^\top \\ 0 & 0 & 0 & 0 \end{pmatrix}^\top, \quad b_i^1 := \left(-\iota_i + r \quad -\iota_i + \frac{r}{2} \quad -\iota_i - \frac{r}{2} \quad -\iota_i - r \right),$$

and

$$W^2 = \Bigg(\underbrace{\frac{2}{r}, -\frac{2}{r}, -\frac{2}{r}, \frac{2}{r}, \frac{2}{r}, -\frac{2}{r}, -\frac{2}{r}, \frac{2}{r}, \ldots, \frac{2}{r}, -\frac{2}{r}, -\frac{2}{r}, \frac{2}{r}}_{4d}, \frac{1}{4} \Bigg).$$

The above realization gives exactly the form in (4). Moreover, we summarize the properties of $g_I^{\text{ind},r}$ as following:

Proposition A.2. *For any $x \in \mathcal{D}_{\text{data}}$ and $y \in \mathbb{R}$, we have*

$$g_I^{\textit{ind},r}(x, y) \begin{cases} = y, & x \in I \text{ and } y \in [0, 4], \\ \leq y, & x \in I \oplus \frac{r}{2} \text{ and } y \in [0, 4], \\ = 0, & \text{otherwise}. \end{cases}$$

Furthermore, we obtain the following properties

1. $L(g_I^{\textit{ind},r}) = 3$,
2. $B(g_I^{\textit{ind},r}) \leq \max\{4, d, 1 + r, 2/r\}$,
3. $K(g_I^{\textit{ind},r}) = 24d + 6$.

A.3. Bounding the approximation error

Firstly, we compute the approximation error of using $g_\epsilon^{f_0}$ to estimate $f_0 = f^* + 2$. Recall that we defined $g_\epsilon^{f_0} := g^{\max,5^d} \circ g^{\text{sum}} \circ g_{\epsilon/2}^{\text{simul}}$. When $x \in S$, there exists some $I \in \mathcal{C}$ such that $x \in I$. Then for this x, we have

$$\begin{aligned} g_\epsilon^{f_0}(x) &= \max_{j \in [5^d]} \sum_{I_k \in \mathcal{C}_j} g_{\epsilon/2,k}^{\text{simul}}(x) \\ &= \max_{I_k \in \mathrm{Neig}(I)} g_{\epsilon/2,k}^{\text{simul}}(x) \\ &\leq \max_{I_k \in \mathrm{Neig}(I)} g_{\epsilon/2,k}^{\text{poly}}(x), \end{aligned}$$

where $\mathrm{Neig}(I) = \{I' \in \mathcal{C} | (I \oplus 3r/2) \cap I' \neq \emptyset\}$ denotes the $3r/2$-neighborhood of hypercube I. In other words, when computing $g_\epsilon^{f_0}(x)$, we only need to take maximum over the estimated function value within hypercubes near x. The second equality follows from the Proposition A.2 that $g_{\epsilon/2,l}^{\text{simul}}(x) = 0$ for $I_l \notin \mathrm{Neig}(I)$ and $d(I_l, I_k) > r$ holds for $I_l \neq I_k \in \mathcal{C}_i$ for all i. The last inequality is due to the construction of $g_{\epsilon/2}^{\text{simul}}$ in (6).

Given $\epsilon \in (0,1)$, we ensure $0 \le g^{\text{simul}}_{\epsilon/2,k}(x) \le 4$ for all $I_k \in \mathcal{C}$ by Proposition A.2, since $g^{\text{simul}}_{\epsilon/2,k}$ approximates f_{I_k} which is an $\epsilon/2$-accuracy Taylor polynomial of $f_0 \in [1,3]$. When $x \in I_t$, the error is bounded as

$$\begin{aligned}
|g^{f_0}_\epsilon(x) - f_0(x)| =& \max\Big\{\max_{I_k \in \mathrm{Neig}(I_t)} g^{\text{simul}}_{\epsilon/2,k}(x) - f_0(x), f_0(x) - \max_{I_k \in \mathrm{Neig}(I_t)} g^{\text{simul}}_{\epsilon/2,k}(x)\Big\} \\
\le& \max\Big\{\max_{I_k \in \mathrm{Neig}(I_t)} g^{\text{poly}}_{\epsilon/2,k}(x) - f_0(x), f_0(x) - g^{\text{poly}}_{\epsilon/2,t}(x)\Big\} \\
\le& \max_{I_k \in \mathrm{Neig}(I_t)} \Big|g^{\text{poly}}_{\epsilon/2,k}(x) - f_0(x)\Big| \\
\le& \max_{I_k \in \mathrm{Neig}(I_t)} \Big|g^{\text{poly}}_{\epsilon/2,k}(x) - f_{I_k}(x)\Big| + \max_{I_k \in \mathrm{Neig}(I_t)} \big|f_{I_k}(x) - f_0(x)\big| \\
\le& \frac{\varepsilon}{2} + d^\beta \left(\frac{3r}{2}\right)^\beta \le \epsilon,
\end{aligned}$$

where the last inequality follows from (5) and Lemma 5.1. In terms of parameter tuning, we choose $r = d^{-1}\epsilon^{1/\beta}/2$.

Next, we extend approximation results of f_0 to f^*. To do so, we firstly implement a neural network $g^{\text{mod}}(z) = (-z + 1) \circ \mathrm{ReLU}(-z+2) \circ \mathrm{ReLU}(z-1)$, which has the equivalent form $g^{\text{mod}}(z) = \min(\max(1,x),3) - 2$ for any $z \in \mathbb{R}$. In addition, g^{mod} has the following properties:

$$L(g^{\text{mod}}) = 3, \;\; B(g^{\text{mod}}) \le 2, \;\; \text{and } K(g^{\text{mod}}) = 12.$$

Then consider $g^{f^*}_\epsilon = g^{\text{mod}} \circ g^{f_0}_\epsilon$ to obtain the desired approximation error ϵ for any $x \in S$

$$\begin{aligned}
\sup_{x \in \mathcal{D}_{\text{data}}} |g^{f^*}_\epsilon(x) - f^*(x)| =& \sup_{x \in \mathcal{D}_{\text{data}}} \Big|\min\big(\max\big(1, g^{f_0}_\epsilon(x)\big), 3\big) - (f^*(x) + 2)\Big| \\
=& \sup_{x \in \mathcal{D}_{\text{data}}} \Big|\min\big(\max\big(1, g^{f_0}_\epsilon(x)\big), 3\big) - f_0(x)\Big| \\
\le& \sup_{x \in \mathcal{D}_{\text{data}}} \Big|g^{f_0}_\epsilon(x) - f_0(x)\Big| \\
\le& \epsilon.
\end{aligned}$$

A.4. Computing network sizes

Recall that the ReLU neural network $g^{f^*}_\epsilon$ is defined as

$$\begin{aligned}
g^{f^*}_\epsilon =& g^{\text{mod}} \circ g^{\max,5^d} \circ g^{\text{sum}} \circ g^{\text{simul}}_{\epsilon/2} \\
=& g^{\text{mod}} \circ g^{\max,5^d} \circ g^{\text{sum}} \circ \big(g^{\text{ind},r}_{I_1} \circ g^{\text{filter}}_1, \ldots, g^{\text{ind},r}_{I_{N_r(S)}} \circ g^{\text{filter}}_{N_r(S)}\big) \circ \big(g^{Id}_{d,L}, g^{\text{poly}}_{\epsilon/2}\big).
\end{aligned}$$

Note that $N_r(S) \le c_0 r^{-p}$. Combined with sub-neural network structures given in Appendix B.1.1 of Nakada & Imaizumi (2020), $g^{f^*}_\epsilon$ has the following properties:

$$\begin{aligned}
L(g^{f^*}_\epsilon(x)) =& L(g^{\text{mod}}) + L(g^{\max,5^d}) + L(g^{\text{ind},r}_{I_1}) + L(g^{\text{filter}}_1) + L(g^{\text{poly}}_{\epsilon/2}) \\
\le& 11 + 2d\log_2 5 + (11 + (1+\beta)/d)(2 + \log_2\beta), \\
B(g^{f^*}_\epsilon(x)) \le& \max\big\{B(g^{\text{mod}}), B(g^{\max,5^d}), B(g^{\text{ind},r}_{I_1}), B(g^{\text{filter}}_1), B(g^{Id}_{d,L}), B(g^{\text{poly}}_{\epsilon/2})\big\}, \\
\le& \max\{d, 1+r, 2/r, c^{\text{poly}}_1 R_S^{\beta s^{\text{poly}}_1} \epsilon^{-s^{\text{poly}}_1}\}, \\
\le& \max\{4d\epsilon^{-1/\beta}, c^{\text{poly}}_1 R_S^{\beta s^{\text{poly}}_1} \epsilon^{-s^{\text{poly}}_1}\}, \\
K(g^{f^*}_\epsilon(x)) \le& 2K(g^{\text{mod}}) + 2K(g^{\max,5^d}) + 2N_r(S) \cdot K(g^{\text{ind},r}_{I_1} \circ g^{\text{filter}}_1) + 2K(g^{Id}_{d,L}) + 2K(g^{\text{poly}}_{\epsilon/2}) \\
\le& 2c^{\text{poly}}_2 R_S^p \epsilon^{-p/\beta} + 2(50d + 17 + c^{\text{poly}}_2)N_r(S) \\
&+ 2(12 + 42 \times 5^d + 2d + 2d(11 + (1+\beta)/p)(2 + \log_2\beta))), \\
\le& 2c^{\text{poly}}_2 R_S^p \epsilon^{-p/\beta} + 2(50d + 17 + c^{\text{poly}}_2)c_0(2d)^p\epsilon^{-p/\beta} \\
&+ 2(12 + 42 \times 5^d + 2d + 2d(11 + (1+\beta)/p)(2 + \log_2\beta))),
\end{aligned}$$

where $c_2^{\mathrm{poly}} = O(d^{2+\lfloor \beta \rfloor})$. By adjusting several constants, we obtain the statement.

B. Proof of Theorem 3.2

The proof of Theorem 3.2 mainly follows Chen et al. (2022). Our Lemma 5.3 and 5.4 is a slight revision of Lemma 5 and 6 in Chen et al. (2022), where we substitute ℓ_∞ covering number with ℓ_2 covering number to deal with unbounded domain. In this section, we compute the ℓ_2 covering number of neural network class and then present the proof of Theorem 3.2.

B.1. Proof of Lemma 5.5

Proof of Lemma 5.5. To construct a covering for $\mathcal{F}(H, L, B, K)$, we discretize each parameter by a unit grid with grid size h. Recall that we write $f \in \mathcal{F}(H, L, B, K)$ as $f(x) = W_L \cdot \mathrm{ReLU}(W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}) + b_L$ in (2). Choose any $f, f' \in \mathcal{F}(H, L, B, K)$ with parameters at most h apart from each other. Denote the weight matrices and intercepts in f, f' as $W_L, \dots, W_1, b_L, \dots, L_1$ and $W'_L, \dots, W'_1, b'_L, \dots, L'_1$ respectively, where $W_l \in \mathbb{R}^{d_l \times d_{l-1}}$ and $b_l \in \mathbb{R}^{d_l}$ for $l = 1, \dots, L$. Without loss of generality, we assume $d_l \leq K$ since all the parameters have at most K nonzero entries. If the input dimension is larger than K, we let the redundant dimensions of input equal to zeros.

Notice that for any random variable $y \in \mathbb{R}^{d_{l-1}}$ which is subject to a distribution P_Y, we have

$$\int_{\mathbb{R}^{d_{l-1}}} \left\| (W_l y + b_l) - (W'_l y + b'_l) \right\|_2^2 \mathrm{d}P_Y(y) = \int_{\mathbb{R}^{d_{l-1}}} \left\| \sum_{i=1}^{d_{l-1}} (W_{l,i} - W'_{l,i}) y_i + (b_l - b'_l) \right\|_2^2 \mathrm{d}P_Y(y).$$

By the inequality $\|t + s\|^2 \leq 2\|t\|^2 + 2\|s\|^2$ which holds for any $s, t \in \mathbb{R}^{d_l}$, we obtain

$$\begin{aligned}\int_{\mathbb{R}^{d_{l-1}}} \left\| (W_l y + b_l) - (W'_l y + b'_l) \right\|_2^2 \mathrm{d}P_Y(y) \leq & 2 \int_{\mathbb{R}^{d_{l-1}}} \left\| \sum_{i=1}^{d_{l-1}} (W_{l,i} - W'_{l,i}) y_i \right\|^2 \mathrm{d}P_Y(y) + 2\|b_l - b'_l\|_2^2 \\ \leq & 2 \sup_{i=1\dots,d_{l-1}} \left\| W_{l,i} - W'_{l,i} \right\|_2^2 \cdot \int_{\mathbb{R}^{d_{l-1}}} \sum_{i=1}^{d_{l-1}} y_i^2 \,\mathrm{d}P_Y(y) + 2\|b_l - b'_l\|_2^2 .\end{aligned}$$

Since parameters W_l, b_l differ at most h from W'_l, b'_l with respect to each entry, we get

$$\begin{aligned}\int_{\mathbb{R}^{d_{l-1}}} \left\| (W_l y + b_l) - (W'_l y + b'_l) \right\|_2^2 \mathrm{d}P_Y(y) &\leq 2 d_{l-1} h^2 \|y\|_{L_2(P_Y)}^2 + 2 d_l h^2 \\ &\leq 2Kh^2 \|y\|_{L_2(P_Y)}^2 + 2Kh^2. \end{aligned} \tag{7}$$

Similarly, we have

$$\begin{aligned}\int_{\mathbb{R}^{d_{l-1}}} \left\| W_l y + b_l \right\|_2^2 \mathrm{d}P_Y(y) =& \int_{\mathbb{R}^{d_{l-1}}} \left\| \sum_{i=1}^{d_{l-1}} W_{l,i} y_i + b_l \right\|_2^2 \mathrm{d}P_Y(y) \\ \leq & 2 \int_{\mathbb{R}^{d_{l-1}}} \left\| \sum_{i=1}^{d_{l-1}} W_{l,i} y_i \right\|^2 \mathrm{d}P_Y(y) + 2\|b_l\|_2^2 \\ \leq & 2 \sup_{i=1\dots,d_{l-1}} \|W_{l,i}\|_2^2 \cdot \int_{\mathbb{R}^{d_{l-1}}} \sum_{i=1}^{d_{l-1}} y_i^2 \,\mathrm{d}P_Y(y) + 2\|b_l\|_2^2 \\ \leq & 2 d_{l-1} B^2 \|y\|_{L_2(P_Y)}^2 + 2 d_l B^2 \\ \leq & 2KB^2 \|y\|_{L_2(P_Y)}^2 + 2KB^2. \end{aligned} \tag{8}$$

Since the ReLU actiavtion function is 1-Lipschitz continuous for each coordinate, we can apply (7) and (8) repeatedly to bound $\|f - f'\|^2_{L_2(P_{\mathrm{data}},S)}$:

$$\begin{aligned}
\|f - f'\|^2_{L_2(P_{\mathrm{data}},S)} =& \int_S \Big| W_L \cdot \mathrm{ReLU}(W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}) + b_L \\
& - W_L' \cdot \mathrm{ReLU}(W_{L-1}' \cdots \mathrm{ReLU}(W_1' x + b_1') \cdots + b_{L-1}') - b_L' \Big|^2 \, \mathrm{d}P_{\mathrm{data}}(x) \\
\leq & 2 \int_S \Big| W_L \cdot \mathrm{ReLU}(W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}) + b_L \\
& - W_L' \cdot \mathrm{ReLU}(W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}) - b_L' \Big|^2 \, \mathrm{d}P_{\mathrm{data}}(x) \\
& + 2 \, \|W_L'\|_2^2 \int_S \Big\| (W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}) \\
& - (W_{L-1}' \cdots \mathrm{ReLU}(W_1' x + b_1') \cdots + b_{L-1}') \Big\|_2^2 \, \mathrm{d}P_{\mathrm{data}}(x) \\
\leq & 4Kh^2 + 4Kh^2 \int_S \|W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}\|^2 \, \mathrm{d}P_{\mathrm{data}}(x) \\
& + 2KB^2 \int_S \Big\| (W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}) \\
& - (W_{L-1}' \cdots \mathrm{ReLU}(W_1' x + b_1') \cdots + b_{L-1}') \Big\|_2^2 \, \mathrm{d}P_{\mathrm{data}}(x).
\end{aligned}$$

Besides, we derive the following bound on $\|W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}\|_{L^2(P_{\mathrm{data}},S)}$:

$$\begin{aligned}
& \int_S \|W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}\|^2 \, \mathrm{d}P_{\mathrm{data}}(x) \\
& \quad \leq 2KB^2 \int_S \|W_{L-2} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-2}\|^2 \, \mathrm{d}P_{\mathrm{data}}(x) + 2KB^2 \\
& \quad \leq (2KB^2)^{L-1} d R_S^2 + (2KB^2)^{L-1} \\
& \quad \leq 2^L (KB^2)^{L-1} d R_S^2,
\end{aligned}$$

where the last inequality is derived by induction and $\|x\|^2 = \sum_{i=1}^d x_i^2 \leq dR_S^2$ for any $x \in S$. Substituting back into the bound for $\|f - f'\|^2_{L_2(P_{\mathrm{data}},S)}$, we obtain

$$\begin{aligned}
\|f - f'\|^2_{L_2(P_{\mathrm{data}},S)} \leq & 4Kh^2 + 2^{L+2} K^L B^{2(L-1)} h^2 d R_S^2 \\
& + 4KB^2 \int_S \Big\| (W_{L-1} \cdots \mathrm{ReLU}(W_1 x + b_1) \cdots + b_{L-1}) \\
& - (W_{L-1}' \cdots \mathrm{ReLU}(W_1' x + b_1') \cdots + b_{L-1}') \Big\|_2^2 \, \mathrm{d}P_{\mathrm{data}}(x) \\
\leq & 4(L-1)(KB^2)^{L-1} h^2 + 2^{L+2} (L-1) K^L B^{2(L-1)} h^2 d R_S^2 \\
& + (2KB^2)^{L-1} \int_S \|W_1 x + b_1 - W_1' x - b_1'\|_2^2 \, \mathrm{d}P_{\mathrm{data}}(x) \\
\leq & 4^{L-1} L K^L B^{2L} h^2 d R_S^2,
\end{aligned}$$

where the second inequality is obtained by induction. Therefore, combining the above inequality with $\|f\|_\infty \leq 1$ for any $f \in \mathcal{F}(L, B, K)$ and $P_{\mathrm{data}}(S^c) \leq \tau$, we get

$$\begin{aligned}
\|f - f'\|^2_{L_2(P_{\mathrm{data}})} =& \int_S |f(x) - f'(x)|^2 \, \mathrm{d}P_{\mathrm{data}}(x) + \int_{S^c} |f(x) - f'(x)|^2 \, \mathrm{d}P_{\mathrm{data}}(x) \\
\leq & 4^{L-1} L K^L B^{2L-2} h^2 d R_S^2 + 4\tau.
\end{aligned}$$

Now we choose h satisfying $h = \sqrt{(\delta^2 - 4\tau)/(4^{L-1} L K^L B^{2L-2} d R_S^2)}$. Then discretizing each parameter uniformly into $2B/h$ grid points yields a δ-covering on $\mathcal{F}(L, B, K)$. Moreover, the covering number $\mathcal{N}_2(\delta, \mathcal{F}(L, B, K))$ satisfies

$$\log \mathcal{N}_2(\delta, \mathcal{F}(L, B, K)) \leq K \log\left(\frac{2B}{h}\right) = K \log\left(\frac{2^L \sqrt{dL} K^{L/2} B^L R_S}{\sqrt{\delta^2 - 4\tau}}\right).$$

□

B.2. Proof of Theorem 3.2

Proof of Theorem 3.2. The square error of the estimator $\widehat{f}$ can be decomposed into a squared bias term and a variance term, which can be bounded using the covering number of the function class. According to Lemmas 5.3 and 5.4, for any constant $\delta \in (0,1)$, we have

$$\begin{aligned}\mathbb{E}\|\widehat{f}-f^*\|^2_{L^2(P_{\mathrm{data}})} \leq & c \inf_{f\in\mathcal{F}(L,B,K)} \|f(x)-f^*(x)\|^2_{L^2(P_{\mathrm{data}})} \\ & + c\sigma^2 \frac{\log\mathcal{N}_2(\delta,\mathcal{F}(L,B,K))+2}{n} \\ & + c\Bigg(\sqrt{\frac{\log\mathcal{N}_2(\delta,\mathcal{F}(L,B,K))+2}{n}}+1\Bigg)\sigma\delta \\ & + \frac{c'}{3n}\log\mathcal{N}_2(\delta/4H,\mathcal{F}(L,B,K))+c'\delta.\end{aligned} \tag{9}$$

Choose $\epsilon = (2dr)^\beta$ in Theorem 3.1. Accordingly, we set tuple (L,B,K) as

$$L = C_1,\ \ B = O(R_S^{\beta s} r^{-s\beta}),\ \ \text{and}\ K = C_2 R_S^p r^{-p}.$$

Then we have

$$\inf_{f\in\mathcal{F}(L,B,K)} \|f(x)-f^*(x)\|^2_{L^2(P_{\mathrm{data}})} \leq (2dr)^{2\beta}+4\tau.$$

Invoking the upper bound of the covering number in Lemma 5.5, we derive

$$\begin{aligned}\mathbb{E}\left\|\widehat{f}-f^*\right\|^2_{L^2(P_{\mathrm{data}})} \leq & (2dr)^{2\beta}+4\tau+\frac{c\sigma^2}{n}\Bigg(K\log\Bigg(\frac{2^L\sqrt{dL}K^{L/2}B^L R_S}{\sqrt{\delta^2-4\tau}}\Bigg)+2c\Bigg) \\ & + c\sqrt{\frac{K\log(2^L\sqrt{dL}K^{L/2}B^L R_S/\sqrt{\delta^2-4\tau})+1}{n}}\sigma\delta \\ & + \frac{c'}{3n}K\log\Bigg(\frac{2^L\sqrt{dL}K^{L/2}B^L R_S}{\sqrt{\delta^2-4\tau}}\Bigg)+(c\sigma+c')\delta \\ = & O\Bigg(\tau+d^{2\beta}r^{2\beta}+\frac{\sigma^2 K}{n}\log\Bigg(\frac{\sqrt{dL}K^{L/2}B^L R_S}{\sqrt{\delta^2-4\tau}}\Bigg) \\ & + \Bigg(\frac{K\log(\sqrt{dL}K^{L/2}B^L R_S/\sqrt{\delta^2-4\tau})}{n}\Bigg)^{1/2}\sigma\delta+\sigma\delta+\frac{1}{n}\Bigg).\end{aligned}$$

By Cauchy-Schwartz inequality, for $0<\delta<1$, we have

$$\mathbb{E}\left\|\widehat{f}-f^*\right\|^2_{L^2(P_{\mathrm{data}})} = O\Bigg(\tau+d^{2\beta}r^{2\beta}+\frac{\sigma^2 K}{n}\log\Bigg(\frac{\sqrt{dL}K^{L/2}B^L R_S}{\sqrt{\delta^2-4\tau}}\Bigg)+\sigma\delta+\frac{1}{n}\Bigg).$$

Plugging in our choice of (L,B,K), we get

$$\mathbb{E}\left\|\widehat{f}-f^*\right\|^2_{L^2(P_{\mathrm{data}})} = O\Bigg(\tau+d^{2\beta}r^{2\beta}+\frac{\sigma^2 d}{n}\left(\frac{R_S}{r}\right)^p\log\Bigg(\frac{(R_S/r)^p}{\delta^2-4\tau}\Bigg)+\sigma\delta+\frac{1}{n}\Bigg)$$

Now we choose $\delta = r^{2\beta}$. Then we deduce the desired estimation error bound

$$\begin{aligned}\mathbb{E}\|\widehat{f}-f^*\|^2_{L^2(P_{\mathrm{data}})} &= O\Bigg(\tau+\sigma r^{2\beta}+\frac{\sigma^2}{n}\left(\frac{R_S}{r}\right)^p\log\Bigg(\frac{(R_S/r)^p}{r^{4\beta}-4\tau}\Bigg)+\frac{1}{n}\Bigg) \\ &= O\Bigg(\tau+\sigma r^{2\beta}+\frac{\sigma^2}{n}\left(\frac{R_S}{r}\right)^p\log\Bigg(\frac{(R_S/r)^p}{r^{4\beta}-4\tau}\Bigg)\Bigg).\end{aligned}$$

The last equality is due to $R_S/r>1$. □

C. Proof of Lemma 4.3

Proof of Lemma 4.3. For simplicity of proof, set $Q = I$. By the construction of $S(R, r; p)$ in (3), we notice that

$$\mathbb{P}(X \notin S(R, r; p)) \leq \mathbb{P}\bigg(\sum_{i=1}^{p} \frac{x_i^2}{\lambda_i^2} > R^2\bigg) + \mathbb{P}\bigg(|x_j| > \frac{r}{2} \text{ for some } j \in [p+1, d]\bigg).$$

Since $X_{1:p} = (x_1, \ldots, x_p) \sim N(0, \Lambda_p)$ where $\Lambda_p = \mathrm{diag}(\lambda_1^2, \ldots, \lambda_p^2)$, the variable $Z = \Lambda_p^{-1/2} X_{1:p} \sim N(0, I_p)$ is a standard Gaussian. Then for any fixed $R > 0$, the probability $\mathbb{P}(\sum_{i=1}^{p} x_i^2/\lambda_i^2 > R^2)$ is equal to $\mathbb{P}(\|Z\|^2 > R^2)$. Moreover, by Lemma E.2, if we choose $R^2 > p$, we will have

$$\begin{aligned}
\mathbb{P}\big(\|Z\|^2 > R^2\big) \leq& \Big(\frac{2R^2 + p}{p}\Big)^{\frac{p}{2}} \exp\Big(-\frac{R^4}{2R^2 + p}\Big) \\
=& \exp\Big(-\frac{p}{2} \cdot \frac{R^4/p^2}{R^2/p + 1/2} + \frac{p}{2} \cdot \log\Big(\frac{2R^2}{p} + 1\Big)\Big) \\
=& O\bigg(\Big[\exp\Big(-\frac{2R^2}{3p} + \log\Big(\frac{3R^2}{p}\Big)\Big)\Big]^{\frac{p}{2}}\bigg).
\end{aligned}$$

Besides, by Lemma E.1, for $j = p+1, \ldots, d$, we derive

$$\mathbb{P}\bigg(|x_j| > \frac{r}{2}\bigg) = \mathbb{P}\bigg(\bigg|\frac{x_j}{\lambda_j}\bigg| > \frac{r}{2\lambda_i}\bigg) = O\bigg(\exp\Big\{-\frac{r^2}{8\lambda_j^2}\Big\}\bigg).$$

Then we can apply the union bound of probability to get

$$\mathbb{P}\bigg(|x_j| > \frac{r}{2} \text{ for some } j \in [p+1, d]\bigg) \leq \sum_{j=p+1}^{d} \mathbb{P}\bigg(|x_j| > \frac{r}{2}\bigg) = O\bigg(\sum_{j=p+1}^{d} \exp\Big(-\frac{r^2}{8\lambda_j^2}\Big)\bigg).$$

Recall that we choose $\lambda_p^{-1} = 2R/r$. Then we have

$$\begin{aligned}
\mathbb{P}\bigg(|x_j| > \frac{r}{2} \text{ for some } j \in [p+1, d]\bigg) =& O\bigg(\sum_{j=p+1}^{d} \exp\Big(-\frac{\lambda_p^2}{2\lambda_j^2} R^2\Big)\bigg) \\
\leq& O\bigg(\sum_{j=p+1}^{d} \exp\Big(-\frac{R^2}{2}\Big)\bigg) \\
=& O\bigg(\exp\Big(-\frac{R^2}{2} + \log(d-p)\Big)\bigg),
\end{aligned}$$

where the inequality comes from $\lambda_j^2 \leq \lambda_p^2$ for $j = p+1, \ldots, d$. Therefore, we have

$$\mathbb{P}(X \notin S(R, r; p)) = O\bigg(\Big[\exp\Big(-\frac{2R^2}{3p} + \log\Big(\frac{3R^2}{p}\Big)\Big)\Big]^{\frac{p}{2}}\bigg) + O\bigg(\exp\Big(-\frac{R^2}{2} + \log(d-p)\Big)\bigg).$$

Next, we compute the covering number of $S(R, r; p)$ using hypercubes with side length $r > 0$, which is denoted as $N_r(S(R, r; p))$. Notice that the first-p-dimensional hyper-ellipsoid of $S(R, r; p)$ is contained in a p-dimensional hyper-rectangle with side length $2\lambda_i R$ for $i = 1, \ldots, p$, while only one hypercube is required to cover the j-th dimension for $j = p+1, \ldots, d$. With this observation, we derive the upper bound for $N_r(S(R, r; p))$:

$$N_r(S(R, r; p)) \leq \prod_{i=1}^{p} \Big(\frac{2\lambda_i R}{r}\Big) = \prod_{i=1}^{p} \Big(\frac{\lambda_i}{\lambda_p}\Big),$$

where the last equality results from our choice of p. □

D. Proof of Theorem 4.5

D.1. Generalization error under exponential eigenvalue decay

Combining the criteria $\lambda_p^{-1} = 2R/r$ and the exponential eigenvalue decay in Assumption 4.1, we have

$$\frac{1}{\mu}\exp(\theta p) = \frac{2R}{r}.$$

Moreover, by Lemma 4.3, we can compute the covering number of $S(R, r, p)$:

$$N_r(S(R, r; p)) \leq \prod_{i=1}^{p}\left(\frac{\lambda_i}{\lambda_p}\right) = \prod_{i=1}^{p}\exp(\theta(p-i)) \leq \exp\left(\frac{\theta p^2}{2}\right) = \left(\frac{2\mu R}{r}\right)^{p/2},$$

which indicates that effective Minkowski dimension of P_{data} is at most $p/2$. Let $r = n^{-(1-\eta)/(2\beta+\sqrt{\log n/\theta})}$ and $R = \log n$ where $\eta \in (0, 1)$ is an arbitrarily small constant. Then we obtain

$$\theta p = \log\left(\frac{2\mu R}{r}\right) = \frac{1}{\theta}\log(2\mu) + \log(\log n) + \frac{(1-\eta)\log n}{2\beta + \sqrt{\log n/\theta}} \leq \frac{2(1-\eta)\log n}{2\beta + \sqrt{\log n/\theta}}.$$

Thereby, we can compute the probability ourside $S(R, r; p)$:

$$\mathbb{P}(X \notin S(R, r; p)) = O\left(\left[\exp\left(-\frac{2R^2}{3p} + \log\left(\frac{3R^2}{p}\right)\right)\right]^{\frac{p}{2}} + \exp\left(-\frac{R^2}{2} + \log(d-p)\right)\right) = O(n^{-\log n/3}).$$

Apply Theorem 3.2 with our choice of R, r and p. Accordingly, the tuple (L, B, K) is set as

$$L = C_1,\ \ B = O\left(n^{\frac{\beta s}{2\beta+\sqrt{\log n/\theta}}}(\log n)^{\beta s}\right),\ \text{ and } K = O\left(n^{\frac{\sqrt{\log n/\theta}}{2\beta+\sqrt{\log n/\theta}}}\right).$$

Then we can get

$$\begin{aligned}
\mathbb{E}\left\|\widehat{f} - f^*\right\|^2_{L^2(P_{\text{data}})} &= O\left(\mathbb{P}(X \notin S(R, r; p)) + \sigma r^{2\beta} + \frac{\sigma^2}{n}\cdot\left(\frac{R_S}{r}\right)^{p/2}\log\left(\frac{(R_S/r)^{p/2}}{r^{4\beta} - 4\tau}\right)\right) \\
&= O\left(\sigma n^{-\frac{2\beta(1-\eta)}{2\beta+\sqrt{\log n/\theta}}} + \frac{\sigma^2}{n}\cdot n^{\frac{(1-\eta)^2\log n/\theta}{(2\beta+\sqrt{\log n/\theta})^2}}\cdot(\log n)^{\frac{(1-\eta)\log n/\theta}{2\beta+\sqrt{\log n/\theta}}}\cdot\frac{\log n/\theta}{2\beta+\sqrt{\log n/\theta}}\cdot\log n\right) \\
&= O\left(\sigma n^{-\frac{2\beta(1-\eta)}{2\beta+\sqrt{\log n/\theta}}} + \sigma^2 n^{-1+\frac{(1-\eta)^2\log n/\theta}{(2\beta+\sqrt{\log n/\theta})^2}+\frac{(1-\eta)\log(\log n)/\theta}{2\beta+\sqrt{\log n/\theta}}}\cdot(\log n)^{3/2}\right).
\end{aligned}$$

The last equality utilizes the fact that $(\log n)^{\log n} = n^{\log(\log n)}$. Furthermore, notice that for sufficiently large n satisfying $\log(\log n)/\sqrt{\theta\log n} \leq \eta$, we have

$$\begin{aligned}
\frac{(1-\eta)^2\log n/\theta}{(2\beta+\sqrt{\log n/\theta})^2} + \frac{(1-\eta)\log(\log n)/\theta}{2\beta+\sqrt{\log n/\theta}} &\leq \frac{(1-\eta)^2\log n/\theta}{(2\beta+\sqrt{\log n/\theta})^2} + \frac{(1-\eta)\eta\sqrt{\log n/\theta}}{2\beta+\sqrt{\log n/\theta}} \\
&\leq \frac{(1-\eta)^2\sqrt{\log n/\theta}}{2\beta+\sqrt{\log n/\theta}} + \frac{(1-\eta)\eta\sqrt{\log n/\theta}}{2\beta+\sqrt{\log n/\theta}} \\
&\leq \frac{(1-\eta)\sqrt{\log n/\theta}}{2\beta+\sqrt{\log n/\theta}}.
\end{aligned}$$

Therefore, we use the above observation to derive the following upper bound for generalization error:

$$\mathbb{E}\left\|\widehat{f} - f^*\right\|^2_{L^2(P_{\text{data}})} = O\left(\sigma^2 n^{-\frac{2\beta(1-\eta)}{2\beta+\sqrt{\log n/\theta}}}(\log n)^{3/2}\right).$$

D.2. Generalization error under polynomial eigenvalue decay

Similarly to last section, we firstly combine the criteria $\lambda_p^{-1} = 2R/r$ and the polynomial eigenvalue decay in Assumption 4.2,

$$p = \left(\frac{2\rho R}{r}\right)^{1/\omega}.$$

Moreover, by Lemma 4.3, we can compute the covering number of $S(R, r, p)$:

$$N_r(S(R, r; p)) \leq \prod_{i=1}^{p}\left(\frac{\lambda_i}{\lambda_p}\right) = \prod_{i=1}^{p}\left(\frac{p}{i}\right)^{\omega} = \left(\frac{p^p}{p!}\right)^{\omega} \leq \left(\frac{p^p}{p^{p/2}}\right)^{\omega} = p^{\omega p/2} = \left(\frac{2\rho R}{r}\right)^{p/2},$$

which indicates that effective Minkowski dimension of P_{data} is at most $p/2$. Let $r = n^{-1/(2\beta+n^{\kappa})}$ and $R = n^{1/(2\omega\beta+\omega n^{\kappa})}$ with $\kappa = (1 + 1/\omega)/\omega$. Then we obtain

$$p = \left(\frac{2\rho R}{r}\right)^{1/\omega} = n^{\frac{(1+1/\omega)/\omega}{2\beta+n^{\kappa}}} = n^{\frac{\kappa}{2\beta+n^{\kappa}}}.$$

Thereby, we can compute the probability outside $S(R, r; p)$:

$$\mathbb{P}(X \notin S(R, r; p)) = O\biggl(\left[\exp\left(-\frac{2R^2}{3p} + \log\left(\frac{3R^2}{p}\right)\right)\right]^{\frac{p}{2}} + \exp\Bigl(-\frac{R^2}{2} + \log(d-p)\Bigr)\biggr) = O\Bigl(\exp\Bigl(-n^{\frac{2}{(2\omega\beta+\omega n^{\kappa})}}/3\Bigr)\Bigr).$$

Apply Theorem 3.2 with our choice of R, r and p. Accordingly, the tuple (L, B, K) is set as

$$L = C_1, \quad B = O\Bigl(n^{\frac{(1+1/\omega)\beta s}{2\beta+n^{\kappa}}}\Bigr), \quad \text{and } K = O\Bigl(n^{\frac{(1+1/\omega)n^{\kappa/(2\beta+n^{\kappa})}}{4\beta+2n^{\kappa}}}\Bigr).$$

Then we have

$$\begin{aligned} \mathbb{E}\left\|\widehat{f} - f^*\right\|_{L^2(P_{\text{data}})}^2 &= O\biggl(\mathbb{P}(X \notin S(R, r; p)) + \sigma r^{2\beta} + \frac{\sigma^2}{n}\cdot\left(\frac{R_S}{r}\right)^{p/2}\log\biggl(\frac{(R_S/r)^{p/2}}{r^{4\beta} - 4\tau}\biggr)\biggr) \\ &= O\biggl(\sigma n^{-\frac{2\beta}{2\beta+n^{\kappa}}} + \frac{\sigma^2}{n}\cdot n^{\frac{1+1/\omega}{2\beta+n^{\kappa}}\cdot\frac{1}{2}n^{\frac{\kappa}{2\beta+n^{\kappa}}}}\cdot n^{\frac{\kappa}{2\beta+n^{\kappa}}}\log n\biggr) \qquad (10) \\ &= O\Bigl(\sigma n^{-\frac{2\beta}{2\beta+n^{\kappa}}} + \sigma^2 n^{-1+\frac{1+1/\omega}{2\beta+n^{\kappa}}\cdot\frac{1}{2}n^{\frac{\kappa}{2\beta+n^{\kappa}}}+\frac{\kappa}{2\beta+n^{\kappa}}}\log n\Bigr). \end{aligned}$$

Notice that

$$\begin{aligned} \frac{1+1/\omega}{2\beta+n^{\kappa}}\cdot\frac{1}{2}n^{\frac{\kappa}{2\beta+n^{\kappa}}} + \frac{\kappa}{2\beta+n^{\kappa}} &= \frac{1+1/\omega}{2\beta+n^{\kappa}}\left(\frac{1}{2}n^{\frac{\kappa}{2\beta+n^{\kappa}}} + \frac{1}{\omega}\right) \\ &\leq \frac{2}{2\beta+n^{\kappa}}\left(\frac{1}{2}n^{\frac{\kappa}{2\beta+n^{\kappa}}} + 1\right) \\ &\leq \frac{n^{\kappa}}{2\beta+n^{\kappa}}, \end{aligned}$$

where the first inequality is due to $\omega > 1$. Therefore, plug the above inequality in (10), we derive the following upper bound for generalization error:

$$\mathbb{E}\left\|\widehat{f} - f^*\right\|_{L^2(P_{\text{data}})}^2 = O\Bigl(\sigma^2 n^{-\frac{2\beta}{2\beta+n^{\kappa}}}\log n\Bigr).$$

E. Auxiliary Lemmas

In this section, we investigate the probability tail bound of standard Gaussian variable, which is useful for the proof of Lemma 4.3. At first, we compute the tail bound for multivariate Gaussian variable.

Lemma E.1. *Suppose $Z = (z_1, \dots, z_p) \sim N(0, I_p)$ is a standard Gaussian variable in $\mathbb{R}^p$. Then for any $t > 0$, we have*

$$\mathbb{P}\big(\|Z\| > t\big) \leq \left(\frac{2t^2 + p}{p}\right)^{\frac{p}{2}} \exp\left(- \frac{t^4}{2t^2 + p}\right).$$

Proof. By the Markov's inequality, for any $\mu \in (0, 1/2)$, we have

$$\begin{aligned}\mathbb{P}\big(\|Z\| > t\big) =& \mathbb{P}\big(\exp(\mu\|Z\|^2) > \exp(\mu t^2)\big) \\ \leq& \frac{\mathbb{E}\exp(\mu\|Z\|^2)}{\exp(\mu t^2)} \\ =& \frac{\prod_{i=1}^p \mathbb{E}\exp(\mu z_i^2)}{\exp(\mu t^2)},\end{aligned}$$

where the last equality comes from the independence of z_i's. To bound $\mathbb{E}\exp(\mu z_i^2)$, we first examine the moment generating function of z_i: for any $t \in \mathbb{R}$,

$$\mathbb{E}\exp(tz_i) = \int_{\mathbb{R}} \exp(tw)\phi(w)\,\mathrm{d}w = \exp(t^2/2),$$

where $\phi(w) = (2\pi)^{-p/2}\exp(-w^2/2)$ denotes the probability density function of stardard Gaussian. Then multiply $\exp(-t^2/(2\mu))$ on both sides,

$$\int_{\mathbb{R}} \exp\left(tw - \frac{t^2}{2\mu}\right)\phi(w)\,\mathrm{d}w = \exp\left(\frac{t^2(\mu - 1)}{2\mu}\right).$$

By integrating both sides with respect to t, we have

$$\sqrt{2\pi\mu}\int_{\mathbb{R}} \exp\left(\frac{\mu w^2}{2}\right)\phi(w)\,\mathrm{d}w = \sqrt{\frac{2\pi\mu}{1-\mu}},$$

which indicates

$$\mathbb{E}\exp(\mu z_i^2) = \mathbb{E}\exp\left(\frac{2\mu z_i^2}{2}\right) = \sqrt{\frac{1}{1 - 2\mu}}.$$

Therefore, for any $\mu \in (0, 1/2)$, we have

$$\mathbb{P}\big(\|Z\| > t\big) \leq (1 - 2\mu)^{-\frac{p}{2}}\exp(-\mu t^2).$$

Let $\mu = t^2/(2t^2 + p)$ and thereby we can conclude the proof of the lemma.

□

For standard Gaussian in $\mathbb{R}$, we derive a tighter upper bound in the following lemma.

Lemma E.2. *Suppose $z \sim N(0, 1)$ is a standard Gaussian variable in $\mathbb{R}$. Then for any $t > 0$, we have*

$$\mathbb{P}\big(\,\|z\| > t\big) \leq \exp\left(-\frac{1}{2}t^2\right).$$

Proof. Firstly, for any $t > 0$, compute the probability that $z > t$:

$$\begin{aligned}\mathbb{P}(z > t) =& \int_t^\infty \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} z^2 \right) \mathrm{d}z \\ =& \int_0^\infty \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} (u+t)^2 \right) \mathrm{d}u \\ =& \exp\left(-\frac{1}{2} t^2 \right) \int_0^\infty \exp(-tu) \cdot \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} u^2 \right) \mathrm{d}u \\ \leq& \exp\left(-\frac{1}{2} t^2 \right) \int_0^\infty \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} u^2 \right) \mathrm{d}u \\ =& \frac{1}{2} \exp\left(-\frac{1}{2} t^2 \right).\end{aligned}$$

Then notice that

$$\mathbb{P}\big(\|z\| > t \big) = \mathbb{P}(z > t) + \mathbb{P}(z < -t) = 2\mathbb{P}(z > t).$$

Thereby, we can conclude the proof. □