Approximation Analysis of Convolutional Neural Networks*

Chenglong Bao[1] Qianxiao Li[2,3] Zuowei Shen[2] Cheng Tai[4]
Lei Wu[5] Xueshuang Xiang[6]

[1]Yau Mathematical Sciences Center, Tsinghua University, China
[2]Department of Mathematics, National University of Singapore,Singapore
[3]Institute of High Performance Computing, A*STAR, Singapore
[4]Beijing Institute of Big Data Research, Peking University, China
[5]School of Mathematical Sciences, Peking University, China
[6]Qian Xuesen Laboratory of Space Technology, China

Abstract

In its simplest form, convolution neural networks (CNNs) consist of a fully connected layer g composed with a sequence of convolution layers T. Although g is known to have the universal approximation property, it is not known if CNNs, which has the form $g \circ T$ inherits this property, especially when the kernel size in T is small. In this paper, we show that under suitable conditions, CNNs does inherit the universal approximation property and its sample complexity can be characterized. In addition, we discuss concretely how the nonlinearity of T can improve the approximation power. Finally, we show that when the target function class has a certain compositional form, convolutional networks are far more advantageous compared with fully connected networks, in terms of number of parameters needed to achieve a desired accuracy.

1 Introduction

Over the past decade, convolutional neural networks (CNNs) have played important roles in many applications, including facial recognition, autonomous driving and disease diagnosis. Such applications typically involve approximating some oracle f^*, which can be a classifier or regressor, by some f chosen from an appropriate model or hypothesis space. In other words, learning involves minimizing the distance between f^* and f over its hypothesis space.

*The work of authors was partially supported by Singapore MOE Research Grant MOE 2014-T2-1-065 and Tan Chin Tuan Centennial Professorship.

Unlike plain fully connected neural networks, convolution neural networks are of the form $f = g \circ T$ where $g \in \mathcal{G}$ is a fully connected classification/regression layer and $T \in \mathcal{T}$ is a feature extractor typically composed of interfacing convolutions and nonlinear activations. From the approximation theory viewpoint, one important direction of investigation is the *universal approximation property* (UAP), namely whether $\{g \circ T : g \in \mathcal{G}, T \in \mathcal{T}\}$ can approximate arbitrary continuous functions on compact domains. The UAP is known to hold in the case of one-hidden-layer, fully connected neural networks for a large class of activation functions [10, 1, 20]. However, for the CNN architecture this is less obvious, even if a fully-connected layer g is present. This is especially so if T consists of convolutions of small filter sizes or the output dimension of T is small, which leads to a loss of information. For example, for classification problems if T maps two samples belong to two different classes into the same feature representation, then it is obvious that no matter what the approximation power of g is, $g \circ T$ cannot correctly classify them. Hence, the first goal of this paper is to show that we can in fact construct CNNs which ensures that when composed with g, forms a universal approximator for classification problems. The key is showing that the convolution-based feature attractors can satisfy the so-called separable condition [24], i.e.

$$T(x_i) - T(x_j) > c, \ \forall x_i \in \Omega_i, x_j \in \Omega_j, i \neq j. \tag{1}$$

for some positive constant c. Here, Ω_i represents the set of samples belonging to class i. Recall that due to small filter sizes and possible dimensional reduction, the satisfaction of this condition for convolution layers is not immediate and the first goal of this paper is to construct convolution feature extractors that satisfy (1) under appropriate sparsity assumptions on the input data, which then allows us to show that a class of practical CNN architectures satisfy the universal approximation property.

Besides the convolutional structure, another important component in CNNs is the non-linear activation function. Commonly used non-linear functions include sigmoid, tanh, and (Leaky) ReLU. These activation functions introduce non-linearity into neural networks and greatly expand their approximation capabilities. In the preceding UAP analysis of CNNs, the effect of non-linearity was not explicitly studied. In fact, in the literature there generally lacks concrete analysis of the advantage of non-linearity, besides general statements such as having a bigger approximation space. In the second part of this paper, we concretely investigate the effect of nonlinear functions in terms of the approximation power by showing that a composition function approximator with non-linear structure can locally improve its approximation, which is not the case for its linear counterpart. More specifically, we establish that if we perform function approximation by composition sequentially, non-linearity allows us to make local progress.

The above analyses demonstrate qualitative approximation properties, but they do not highlight the advantage of CNNs over traditional architectures, such as deep fully connected neural networks. Moreover, the role of depth is not explicitly considered. Therefore, the last important component of CNNs we discuss is the hierarchical structure, which underlies the success of deep learning over shallow neural networks in many complex tasks, e.g. image classification, natural language processing. In practice, it has been shown that the composi-

tional structure in CNNs progressively extracts useful information. However, understanding its theoretical advantage is a largely unsolved problem. Motivated by multi-scale analysis, here we elucidate the interplay between hierarchical structure and model complexity. More concretely, we show that if we assume the target oracle f^* has a compositional form, then a significantly less number of parameters is required for CNNs to approximate it as compared to their fully connected counter-parts. In fact, the reduction is exponential in the number of composition levels, implying that deep convolutional networks are far more advantageous than deep fully connected networks for approximating compositional functions.

We close this section by discussing some related work to the current paper. Understanding how convolutional neural networks work is one of the central open problems in the community of deep learning. Many researchers have attempted to answer this question from various perspectives. While no complete solution has been developed so far, each attempt extends our understanding of the internal mechanism of convolutional neural networks. In the following context, we mainly review the existing works closely related to this paper.

One of the classical results for neural networks is the universal approximation property (UAP) of shallow networks [10, 1, 20], i.e. a neural network with one hidden layer can approximate any continuous function defined on a compact set to arbitrary precision with enough hidden units. More recently, non-asymptotic analysis of the relationship between approximation errors and number of neurons in multi-layer neural networks has been developed [**?**]. More abstractly, the approximation of functions by composition was investigated in [**?**], in which a multi-layer neural networks serve as a means of numerical implementation. Despite the theoretical guarantee, deep fully connected networks are seldomly used in practice. More sophisticated structures (e.g. multi-layer convolutional networks) are preferred, and often yield surprisingly good performance. Therefore, in recent years, many works have attempted to analyze the approximation properties for the multi-layer networks. The approximation ability of convolutional networks has been numerically demonstrated by a series of numerical experiments including randomly corrupting the pixels and labels [39]. In [18, 19, 27], it is shown that the restricted Boltzmann machine and deep belief networks can approximate any distribution on the set $\{0,1\}^n$ of binary vectors of length n. Meanwhile, the width bounded fully connected networks can approximate approximate any continuous function as the depth goes to infinity [22] and the ResNet with one-neuron hidden unit per layer can approximate any Lebesgue-integrable functions [21]. However, the above results do not apply for CNNs. In [8, 7], convolutional arithmetic circuits and convolutional rectified networks are constructed and analyzed via tensor decomposition. Despite the UAP and depth efficiency of these networks, it only contains the 1×1 convolution in each hidden layer which is not consistent with the practical CNNs. [40] constructs a CNN with zero boundary condition and establishes its UAP and convergence rate for the functions in Sobelev space. Compared to the existing CNNs, every hidden layer in [40] contains only one convolution and there is no fully connected layer. Moreover, due to the zero boundary condition, the dimension in feature space is larger than the dimension of input signal especially when the depth of the network is deep. Instead of constructing a new architecture, a main goal of this paper is to understand why the current state-of-the-art CNNs can achieve high classification

accuracy. As the last layer always contains a fully connected layer which has the UAP, our efforts have been made to construct a CNN with filter size equal to 3 such that the features satisfy the separable condition (1). More specifically, checking this condition can be difficult when the feature dimension is less than the input dimension.

Another line of research for understanding deep neural networks is investigating the advantage of non-linearity and multi-layer structures, which are two key ingredients contributing to the success of deep learning. An early attempt is initiated by [23] in which a wavelet transformation based scattering network is constructed, and its translation and deformation invariance are proved. The above properties are generalized to general convolutional filters, Lipschitz non-linear activations and pooling functions in [38]. In [32], a sparsely-connected depth-4 neural network is constructed to approximate functions defined on a low dimensional manifold. Convolutional sparse coding based neural networks, which can been seen as a generalization of data driven tight frame first introduced in [3], has been recently proposed with stable recovery properties under certain sparsity assumptions [29]. The distance-preservation property has been shown in [12] for neural networks with Gaussian random weights. Although these works provide insights into certain compositional architectures, they do not directly lead to understanding good classification performance of CNNs. The same is true for a series of works investigating the role of composition in function approximation [25, 26, 30, 28, 36], in that there lacks concrete results on the advantage of composition for the CNN architectures that are employed in practice. The goal of the present work is to address this issue, and develop approximation results that applies to modern convolution neural network structures.

The rest of this paper is organized as follows. The approximation properties and scaling analysis of convolutional neural networks are shown in section 2 and section 3, respectively. Finally, section 4 concludes and the detailed proofs are shown in Appendix.

2 Approximation Analysis of Convolutional Neural Networks

In this section, we present our approximation result for CNNs following a statistical learning framework [37, 31]. This consists of estimating the so-called bias and variance, which will be made clear subsequently. The bias measures the approximation error, which is the distance between the oracle classifier and the best approximation in the function space generated by convolutional networks. The variance measures the sample error which is the distance between and the classifier obtained by minimizing some empirical loss of examples sampled from an unknown distribution. The key is to prove that there exist convolutional networks which are separable, stable feature extractors, in the sense of (1). The sample error analysis then follows from the classical PAC-learning framework. Before proceeding to the analysis, we first introduce some assumptions and notations.

2.1 Notations and definitions

To avoid the cumbersomeness of notations, we only consider one dimensional signals (vectors) in this paper. Remarks will be provided when the result is dependent on the dimensional of the input signals. Vectors, matrices and sets are denoted by lower, upper and calligraphic letters, respectively. Given a vector $y \in \mathbb{R}^n$, y_j denotes the j-th entry; given a matrix $Y \in \mathbb{R}^{m\times n}$, Y_j denotes the j-th column of Y and Y_{ij} denotes the i-th element in Y_j. The multi-valued functions are denoted by the bold upper letters.

Definition 1. *Consider the following different type of convolution.*

1. *For $u \in \mathbb{R}^n$, $v \in \mathbb{R}^r$, the cyclic convolution $* : \mathbb{R}^n \times \mathbb{R}^r \mapsto \mathbb{R}^n$ is*

$$(u * v)_i = \sum_{k=1}^{r} u_{i\ominus k} v_k,$$

   where $\ominus$ is $i \ominus j = i - j \mod n$;

2. *For $U \in \mathbb{R}^{n\times s}, V \in \mathbb{R}^{r\times s}$, the multi-channel convolution $\circledast : \mathbb{R}^{n\times s} \times \mathbb{R}^{r\times s} \mapsto \mathbb{R}^n$ is*

$$U \circledast V = \sum_{i=1}^{s} U_i * V_i.$$

3. *A 1-layer convolution with m kernels $\{U^i\}_{i=1}^m \subset \mathbb{R}^{r\times s}$ is a nonlinear map $\mathbf{F} = (F^1, F^2, \ldots, F^m)$ where*

$$F^i(X) = X \circledast U^i + B_i, \quad i = 1, 2, \ldots, m,$$

   where B_i is the so-called bias. Given an activation function σ, define $\mathbf{F}_\sigma = (F^1_\sigma, F^2_\sigma, \ldots, F^m_\sigma)$ where $F^i_\sigma(\mathbf{x}) = \sigma(X \circledast U^i + B_i)$, $i = 1, 2, \ldots, m$.

2.2 Problem formulation

We now introduce the basic formulation of the classification problem using CNNs. Although the following analysis can be extended to the multi-classification tasks, we consider the binary case for the simplicity. Let Ω_0 and Ω_1 to be the sets containing the signals from two classes. Throughout this paper, we make the following assumption on Ω_0 and Ω_1.

Assumption 1. *Let $\Omega = \Omega_0 \cup \Omega_1$ where Ω_0 and Ω_1 are compact subsets in $\mathbb{R}^n$. Moreover, there is a positive a gap between Ω_0 and Ω_1. That is, there exists some $d_0 > 0$ such that*

$$d(\Omega_0, \Omega_1) = \inf\{\|x_0 - x_1\| | x_0 \in \Omega_0, x_1 \in \Omega_1\} = d_0 > 0.$$

By the Assumption 1, there exists an oracle classifier $f^* : \Omega \mapsto [0, 1]$ such that

$$f^*(x) = 0, \quad \text{if } x \in \Omega_0; \quad \text{and} \quad f^*(x) = 1, \quad \text{if } x \in \Omega_1. \tag{2}$$

Observe that f^* is continuous due to the compactness of Ω_0 and Ω_1. Given m training samples $\{x_i, f^*(x_i)\}_{i=1}^m$, the classification task aims to find an approximation scheme to obtain a classifier that is close to the oracle classifier. Due to the complicated geometry and the limited information of the domain Ω, it is difficult to find a good approximated classifier using the traditional interpolation schemes. On the contrary, CNNs construct the approximation via the composition of feature map T and classifier g. The feature map T aims to simplify the domain Ω so as to identify Ω_0 and Ω_1 more easily in the feature space. For instance, $T(\Omega_0)$ and $T(\Omega_1)$ may become easily separated. In fact, the feature map T in CNNs is constructed in the form of multi-layer convolutions:

$$T = \mathbf{F}_{L,\sigma} \circ \cdots \circ \mathbf{F}_{1,\sigma},$$

where L is the number of layers and $\mathbf{F}_i$ is a 1-layer convolution with s_i kernels $i = 1, \ldots, L$. The classifier g in CNNs is a one-layer fully connected network with K hidden units:

$$g(x) = \sum_{i=1}^{K} c_i \sigma(w_i^\top x + b_i),$$

where $c_i, b_i \in \mathbb{R}$, $w_i \in \mathbb{R}^n$ and σ is a nonlinear function. The parameters Θ_T and Θ_G of the feature map T and classifier g are

$$\begin{aligned}\Theta_T^L &= \cup_{i=1}^L \Theta_T^i = \cup_{i=1}^L \cup_{j=1}^{s_i} \{U^{ij}, B^{ij}\},\\ \Theta_G^K &= \cup_{i=1}^K \Theta_G^i = \cup_{i=1}^K \cup_{j=1}^{i} \{c_i, b_i, w_i\}.\end{aligned}$$

Throughout this paper, we assume the size of filters is 3 which are most used in practice. In summary, the space $\mathcal{H}^{L,K}$ of the L-layer CNNs is

$$\begin{aligned}\mathcal{T}^L &= \{T = \mathbf{F}_{L,\sigma} \circ \cdots \circ \mathbf{F}_{1,\sigma}(x;\theta_T) | \theta_T \in \Theta_T^L\},\\ \mathcal{G}^K &= \{g(x;\theta_g) = \sum_{i=1}^{K} c_i \sigma(w_i^\top x + b_i) | \theta_g \in \Theta_G^K\},\\ \mathcal{H}^{L,K} &= \{h = g \circ T(x;\theta_T,\theta_G) | T \in \mathcal{T}^L, g \in \mathcal{G}^K\}.\end{aligned} \tag{3}$$

Given m training samples $\{x_i, f^*(x_i)\}_{i=1}^m$, the approximating classifier is obtained via solving the empirical minimization:

$$\min_{\theta_T \in \Theta_T^L, \theta_g \in \Theta_G^K} \frac{1}{m} \sum_{i=1}^{m} (g \circ T(x_i;\theta_T,\theta_g) - f^*(x_i))^2.$$

Define $\Theta^* = (\Theta_T^*, \Theta_G^*)$ be the set of minimizers and assume $\Theta^* \neq \emptyset$. Let $g^* = g(x;\Theta_G^*)$ and $T^* = T(x;\Theta_T^*)$ where $(\Theta_T^*, \Theta_G^*) \in \Theta^*$, the classification accuracy of $g^* \circ T^*$ is

$$\|g^* \circ T^* - f^*\| = \left\{ \int_\Omega (f^*(x) - g^*(T^*(x)))^2 d\mu \right\}^{1/2}, \tag{4}$$

where μ is a probability measure on Ω.

2.3 Overview of the analysis

Direct estimation of (4) is difficult due to the additional complication of sampling errors, and thus most existing approaches estimate (4) by separating the it from the approximation error via the triangle inequality,

$$\|g^* \circ T^* - f^*\| \leq \|g \circ T - f^*\| + \|g \circ T - g^* \circ T^*\|, \quad \forall h = g \circ T \in \mathcal{H}^{L,K}. \tag{5}$$

The first term in the right hand part of (5) is the *bias* which characterizes the approximation power of the space $\mathcal{H}^{L,K}$; the second term is the *variance* which characterizes the errors due to the sampling process and minimization models. In the following context, we analyze the bias and variance separately.

Bias estimation. The *bias* estimation $\|g \circ T - f^*\|$ is to show the approximation capability of the CNNs, i.e. T has deep convolutional architecture, with interlacing convolutional and point-wise non-linear layers. Our goal is to make rigorous statement that convolutional network classifier can approximate the oracle classifier f^* up to any precision.

In the process, we will require a classical result on the approximation properties of full connected networks [10, 15], which is stated below.

Theorem 1. *Let σ be a non-constant, bounded and monotonically-increasing continuous function. Then, $\mathcal{G} = \cup_{K=1}^{\infty} \mathcal{G}^K$ is dense in $C(\mathcal{X})$ for any compact subset $\mathcal{X}$ with respect to the L^∞-norm where $\mathcal{G}^K$ is defined in* (3) *for each $K \in \mathbb{N}$.*

Although Theorem 1 provides theoretical guarantee of the approximation properties of fully connected networks, a similar result for the CNNs cannot be derived as a consequence, due to the presence of feature map T, especially when the range of T (feature space) is low dimensional. In particular, it is clear that $g \circ T$ cannot distinguish two points from different classes when T maps them the same feature.

We now introduce a condition below which eliminates this issue, and from it, the approximation properties of CNNs can be readily established.

Condition 1. *The feature map $T : \mathbb{R}^n \mapsto \mathbb{R}^p$ satisfies the properties as follows.*

1. *T is stable if there exists $L > 0$ such that*

$$\|T(x) - T(y)\| \leq L\|x - y\|, \quad \forall x, y \in \Omega. \tag{6}$$

2. *T is separable if there exists $\ell > 0$ such that*

$$\|T(x_0) - T(x_1)\| \geq \ell, \quad \forall x_0 \in \Omega_0, x_1 \in \Omega_1. \tag{7}$$

The first property mainly ensures that in-class variations are small and the second property ensures that the classifier has enough discriminative power to separate the two classes. We stress that this condition, if true, will lead to the desired approximation results for CNNs. However, it is not obvious that this condition (especially (7)) holds, since T often maps Ω to a lower dimensional space and thus possesses a large "kernel".

In the following, we first present the approximation results which follow from Condition 1, and then show that we can in fact construct CNNs with the usual architecture that satisfies this condition.

Theorem 2. *Let f^* be the oracle classifier in* (2)*. Suppose the Assumption 1 holds and there exists $T \in \mathcal{T}^L$ that satisfies Condition 1 where $\mathcal{T}^L$ is the L-layer convolutional networks defined in* (3)*. Then, for any $\epsilon > 0$, there exist $K \in \mathbb{N}$ and $g \in \mathcal{G}^K$ such that*

$$\|f^* - g \circ T\|_\infty \leq \epsilon,$$

where $\mathcal{G}^K$ is 1-layer fully connected networks defined in (3)*.*

Proof. Let $\hat{\Omega}_i = T(\Omega_i)$ for $i = 0, 1$. By the stableness of T, we know $\hat{\Omega}_0$ and $\hat{\Omega}_1$ are disjoint compact subsets. Define $\hat{f} : \mathbb{R}^p \mapsto [0, 1]$ such that $\hat{f}(x) = i$ for $x \in \hat{\Omega}_i$, for $i = 0, 1$. Then, $\hat{f}$ is continuous and $\hat{f} \circ T = f^*$ for all $x \in \Omega$. Moreover, by the Theorem 1, for any $\epsilon > 0$, there exist $K \in \mathbb{N}$ and $g \in \mathcal{G}^K$ where $\mathcal{G}^K$ is defined in (3) such that

$$\|g - \hat{f}\|_\infty \leq \epsilon.$$

Thus, we have $\|g \circ T - f^*\|_\infty = \|g \circ T - \hat{f} \circ T\|_\infty \leq \epsilon$. □

Variance estimation. The variance term $\|g^* \circ T^* - g \circ T\|$ characterizes the uncertainty between the numerical and the theoretical classifier. A common assumption is imposed for the training samples.

Assumption 2. *The m samples $\{x_i\}_{i=1}^m$ are identically and independently drawn according to a probability measure μ on Ω.*

Given a classifier $f : \Omega \to \{0, 1\}$, define the *error* and *empirical error* function as

$$\mathbb{E}(f) = \int_\Omega (f(x) - f^*(x))^2 d\mu(x), \quad \mathbb{E}_z(f) = \frac{1}{m} \sum_{i=1}^m (f(x_i) - f^*(x_i))^2.$$

The sample error is to estimate the error from the observation of the empirical error which is based on the concentration inequality [9, 4].

Lemma 1 (Bernstein Inequality)**.** *Suppose a random variable ξ on X satisfies $E\xi = \nu \geq 0$, and $|\xi - \nu| \leq B$ almost everywhere. Assume that $E\xi^2 \leq \eta E\xi$. Then for any $\delta > 0$ and $0 < \gamma \leq 1$, we have*

$$\mathbb{P}\left\{ \frac{\nu - \frac{1}{m}\sum_{i=1}^m \xi(z_i)}{\sqrt{\nu + \delta}} > \gamma\sqrt{\delta} \right\} \leq \exp\left\{ -\frac{\gamma^2 m\delta}{2\eta + \frac{2}{3}B} \right\}$$

For any space $\mathcal{H}$ and $\delta > 0$, define $\mathcal{N}_\mathcal{H}(\delta)$ to be the covering number of $\mathcal{H}$, i.e. the minimal number of the balls with radius δ that covers $\mathcal{H}$. By the Lemma 1, we estimate the relationship between the error $\mathbb{E}(f)$ and the empirical error $\mathbb{E}_z(f)$ in the next theorem.

Theorem 3. *Suppose the Assumption 2 holds. If $|f| \leq M$ for all $f \in \mathcal{H}^{L,K}$, then for any $\delta > 0$ and $0 < \gamma < 1$, we have*

$$\mathbb{P}\left\{\sup_{f\in\mathcal{H}^{L,K}} \frac{\mathbb{E}(f) - \mathbb{E}_z(f)}{\sqrt{\mathbb{E}(f)+\delta}} > 4\gamma\sqrt{\delta}\right\} \leq \mathcal{N}_{\mathcal{H}^{L,K}}\left(\frac{\gamma\delta}{2M}\right)\exp\left\{-\frac{3\gamma^2 m\delta}{8}\right\}.$$

The proof is deferred to the appendix A. From (8), it is clear that the probability goes to 0 as $m \to +\infty$ if the covering number is finite. In general, the covering number increase exponentially as the dimension of signal increases which is the curse of dimension. Many efforts have been made to calculate it more precisely with careful analysis [4]. The estimation of covering number for CNNs with specific architecture is an interesting problem in our future research.

Classification accuracy. Once both bias and sample error is estimated, the classification accuracy of the numerical classifier could be obtained. Given m training samples $\{(x_i, f^*(x_i))\}$, recall the numerical classifier $h^* = g^* \circ T^*$ is the minimizer of the empirical error, i.e.

$$(g^*, T^*) \in \arg\min_{g\in\mathcal{G}^K, T\in\mathcal{T}^L} \mathbb{E}_z(g \circ T). \tag{9}$$

Define the classification accuracy of h^* is

$$\mathcal{A}(h^*) = 1 - \mu(\{x \in \Omega | g^* \circ T^*(x) \neq f^*(x)\}) \tag{10}$$

where the second term in (10) measures the incorrectness of the learned classifier h^*. In summary, we can characterize $\mathcal{A}(h^*)$ in the next theorem.

Theorem 4. *Suppose the Assumption 1 and 2 hold. Assume the space $\mathcal{T}^L$ generated by the CNNs satisfies Condition 1. Then, for any $\epsilon > 0$, there exist $K \in \mathbb{N}$ and $M > 0$ such that*

$$\mathbb{P}\{\mathcal{A}(g^* \circ T^*) \geq 1 - \epsilon\} \geq 1 - \mathcal{N}_{\mathcal{H}^{L,K}}(\epsilon/32M)\exp\{-3m\epsilon/256\}, \tag{11}$$

where m is the number of samples.

Proof. The boundedness of $f \in \mathcal{H}^{L,K}$ can be achieved by imposing the bounded constraints on the parameter space. Without loss of generality, we assume $\mathcal{H}^{L,K} = \{g \circ T | \|\theta\|_\infty \leq \tilde{M}\}$ such that $|f| \leq M$ for all $f \in \mathcal{H}^{L,K}$.

By the Theorem 2, there exist a convolutional neural network $T \in \mathcal{T}^L$ and $g \in \mathcal{G}^K$ such that

$$\|g \circ T - f^*\|_\infty \leq \sqrt{\epsilon}/2 \tag{12}$$

Using the sample error bound (8) in Theorem 3,

$$\mathbb{E}(h^*) \leq 4\gamma\sqrt{\delta}\sqrt{\mathbb{E}(h^*)+\delta} + \mathbb{E}_z(h^*) \tag{13}$$

holds with probability at least $1 - \mathcal{N}_{\mathcal{H}^{L,K}}(\frac{\gamma\delta}{2})\exp\left\{-\frac{3\gamma^2 m\delta}{8}\right\}$. The inequality (13) implies

$$\mathbb{E}(h^*) \leq 4\gamma\sqrt{\delta(\mathbb{E}(h^*)+\delta) + \mathbb{E}^2(h^*)/4} + \mathbb{E}_z(h^*) = 4\gamma(\mathbb{E}(h^*)/2 + \delta) + \mathbb{E}_z(h^*)$$

By (9), $h^* = g^* \circ T^*$ is the minimizer of $\mathbb{E}_z(g)$ in $\mathcal{H}^{L,K}$, we have

$$\mathbb{E}_z(h^*) \leq \mathbb{E}_z(g \circ T) \leq \varepsilon/4$$

from (12). Thus, we know

$$\mathbb{E}(g^* \circ T^*) \leq (1-2\gamma)^{-1}(4\gamma\delta + \varepsilon/4)$$

holds with probability at least $1 - \mathcal{N}_{\mathcal{H}^{L,K}}(\frac{\gamma\delta}{2M})\exp\left\{-\frac{3\gamma^2 m\delta}{8}\right\}$. Choose $\gamma = 1/4$ and $\delta = \epsilon/4$, the inequality (11) holds. □

Theorem 4 established that the CNNs can achieve the desired classification with high probability whenever the number of samples is large enough and the Condition 1 holds. However, verifying the stable and separable properties of the feature map T generated by the CNNs is difficult in general. Since the feature map T is continuous, the stable condition (6) is easily verified as Ω is compact. The most technical part is to find a feature map T generated by some CNN architecture such that the separable condition (7) holds. In the next section, we will focus on this part.

2.4 The Separable Property of CNNs

Recall that the feature map $T : \mathbb{R}^n \mapsto \mathbb{R}^p$ is

$$T = \mathbf{F}_{L,\sigma} \circ \cdots \circ \mathbf{F}_{1,\sigma},$$

where $\mathbf{F}_{\mathbf{i},\sigma}$ is given in Definition (1) for all $i = 1, 2, \ldots, L$. Throughout this section, we assume the activation function σ is a ReLU function, i.e. $\sigma(x) = \max(x, 0)$. By considering the dimension of features, we will discuss the separable property in two cases.

2.4.1 Case I: $p \geq n$

In the next lemma, we show that a 1-layer convolution network can represent a wavelet tight frame transformation.

Lemma 2. *For any positive integer $J > 1$, there exist J kernels $\{u^i\}_{i=1}^J \subset \mathbb{R}^J$ such that the 1-layer convolutional network $\mathbf{F} : \mathbb{R}^{n\times 1} \mapsto \mathbb{R}^{n\times J}$, $\mathbf{F} = (F^1, F^2, \ldots, F^J)$ with kernel $(u^1, u^2, \ldots, u^J)$ induces a tight frame.*

Proof. Constructing one example suffices to prove existence. Let $U \in \mathbb{R}^{J\times J}$ be an orthogonal matrix with $U^\top U = \frac{1}{J} I$. Let u^i be the i-th row of U. Define $\mathbf{F} : \mathbb{R}^n \mapsto \mathbb{R}^{nJ}$, which is defined as

$$\mathbf{F}(x) = (u^1 * x; u^2 * x, \ldots, u^J * x).$$

Then, the columns of $\mathbf{F}^\top$ forms a tight frame, i.e. $\mathbf{F}^\top\mathbf{F} = I$. See [3, 35] for more details. □

The transformation $\mathbf{F}$ is isometric, hence separable. In practical convolutional networks, point-wise nonlinearity is shown to be essential. The next theorem shows that convolutional neural networks can represent separable maps when the network employs a nonlinear activation function σ.

Theorem 5. . *For any positive integer $J > 1$, there exists a 1-layer convolutional network $\mathbf{F}_\sigma : \mathbb{R}^{n\times 1} \mapsto \mathbb{R}^{n\times 2J}$ which is a separable and stable feature extractor.*

Proof. Let $\mathbf{F}$ be defined in Lemma 2, then we have

$$\|\mathbf{F}(x) - \mathbf{F}(y)\|^2 = \|x - y\|^2, \quad \text{for } x \in \Omega_0, y \in \Omega_1.$$

Define $\mathbf{F}_\sigma : \mathbb{R}^{n\times 1} \mapsto \mathbb{R}^{nJ\times 2}, \mathbf{F}_\sigma(x) = [\sigma(\mathbf{F}(x)), \sigma(-\mathbf{F}(x))]$, then

$$\begin{aligned}\|\mathbf{F}_\sigma(x) - \mathbf{F}_\sigma(y)\|^2 &= \|\sigma(\mathbf{F}(x)) - \sigma(\mathbf{F}(y))\|^2 + \|\sigma(-\mathbf{F}(x)) - \sigma(-\mathbf{F}(y))\|^2 \\ &\geq \frac{1}{2}\|\mathbf{F}(x) - \mathbf{F}(y)\|^2 = \frac{1}{2}\|x - y\|^2 \geq \frac{1}{2}d.\end{aligned}$$

where the first inequality is from the fact

$$|\sigma(a) - \sigma(b)|^2 + |\sigma(-a) - \sigma(-b)|^2 \geq \frac{1}{2}(a - b)^2 \quad \text{for } a, b \in \mathbb{R}. \tag{14}$$

Thus, $\mathbf{F}_\sigma$ is separable. □

The above 1-layer feature map $T = \mathbf{F}_\sigma : \mathbb{R}^{n\times 1} \mapsto \mathbb{R}^p$, $p = 2nJ$, is sufficient for proving existence of T via increasing the dimensions. However, we use stable feature extractors that has fewer output dimensions than the input in many applications. For example, in AlexNet [17], the convolutional network component has an output dimension of 4096, which is much smaller than the input image dimension 224×224.

2.4.2 Case II: $p < n$

When the dimension of the input signal is so large that it exceeds the available computing resources, dimension reduction is often used, where T extracts a number of features smaller than the dimension of the input space. Such processes often lose information. However, in the case when the input data have some sparsity structures, dimensional reduction can be achieved without sacrificing separability. In particular, sparsity allows a random projection to be nearly isometric with high probability on a lower dimensional space [5], thereby enforcing the separability condition. We will show subsequently that this random projection can be decomposed into small convolutions, consistent with the usual architectures in CNNs.

Since the decomposition results in low dimensional feature, we define valid and full convolutions which naturally result in varying dimensions.

$$\begin{aligned}\text{(Valid)} \quad &*_V : \mathbb{R}^n \times \mathbb{R}^r \mapsto \mathbb{R}^{n-r+1} : \ (u *_V v)_i = (u * v)_{r-i+1}, \\ \text{(Full)} \quad &*_F : \mathbb{R}^n \times \mathbb{R}^r \mapsto \mathbb{R}^{n+r-1} : \ u *_F v = I(u) *_V v.\end{aligned}$$

where $I(u) = [\mathbf{0}, u, \mathbf{0}]^\top \in \mathbb{R}^{n+2r-2}$ and $\mathbf{0}$ is a zero vector in $\mathbb{R}^{r-1}$. A straightforward relationship between valid and full convolution is:

$$x *_V (w_1 *_F w_2) = (x *_V w_1) *_V w_2(-\cdot),$$

where $w(-\cdot)$ is the flip of w_2. In the following context, we begin with a concrete notation of sparsity on which the subsequent results are based.

Definition 2. *x is said to be s-sparse if the number of non-zero elements of x is less than or equal to s. Denote Σ_s be the set of all s-sparse vectors.*

From the classical literature [6] in compressed sensing, we have the following result regarding to s-sparse vectors.

Theorem 6. *For any $\delta_s \in (0, 1)$, there exists a linear map $A : \mathbb{R}^n \mapsto \mathbb{R}^p$ that satisfies the (s, δ_s)-Restricted Isometry Property (RIP):*

$$(1 - \delta_s)\|x\|^2 \leq \|Ax\|^2 \leq (1 + \delta_s)\|x\|^2, \ \forall x \in \Sigma_s.$$

if $p \geq 2C\delta_s^{-2} s \ln(en/s)$ for some constant $C > 0$ which is independent of δ_s and s.

Remark 1. *Among many, one way to pick A to satisfy the above is Gaussian random matrices with i.i.d. entries $\mathcal{N}(0, \frac{1}{p})$. Then, by Theorem 9.27 in [11], there exists $C \approx 80.098$ such that A satisfies (s, δ_s)-RIP with probability at least $1 - 2(en/s)^{-s}$ when $p \geq 2C\delta_s^{-2} s \ln(en/s)$.*

The linear map A constructed above is injective when restricted to the set of s-sparse vectors. Therefore, if the signal itself is sparse, all that remains is to show that A can be decomposed into a sequence of small convolution (this is done in Theorem 8), leading to our desired result.

However, in most applications the signals are not sparse in the temporal and spatial domain. Nevertheless, they often admit a sparse approximation under some transformation, e.g., natural images often admit sparse approximations in wavelet tight frames, and audio signals often have sparse approximations using the orthonormal Fouirer transform. We now make this notion of approximate sparsity precise, which allows for greater applicability of our results.

Definition 3. *(Approximate Sparsity) Given $s \in \mathbb{N}$, Ω is s-approximately-sparse with error $\beta \geq 0$ under a linear transformation W if*

$$\|H_s(Wx) - Wx\| \leq \beta, \quad \forall x \in \Omega,$$

where $H_s(x) = \arg\min\{\|x - y\| : \|y\|_0 \leq s\}$.

Note that $\Omega = \Sigma_s$ if $\beta = 0$. Since W is to be composed with A, the separability condition is preserved if W satisfies $W^\top W = I$. Moreover, we would like to show that W can be

represented by convolution filters. The following outlines a possible construction: Let $k \in \mathbb{N}$ and $n/k = l$. Define the downsample operator $P_k : \mathbb{R}^n \mapsto \mathbb{R}^\ell$ to be $P_k(x)[j] = x[kj]$ for $j = 1, \ldots, l$. Given a kernel w, define $*_k$ be the convolution of stride k, it is easy to know

$$w *_k x = P_k(x * w).$$

Given J kernels $(w_1, \ldots, w_J)$, denote the operator $W : \mathbb{R}^n \mapsto \mathbb{R}^m$ with $m = lJ$ to be

$$W(x) = (x *_k w^1, x *_k w^2, \ldots, x *_k w^J). \tag{15}$$

The requirement $W^\top W = I$ can be satisfied using the unitary extension principle. Applying Theorem 6 for $W(x)$, it has that for any $\delta_{2s} \in (0, 1)$, there exists $A \in \mathbb{R}^{\tilde{p} \times m}$ such that

$$(1 - \delta_{2s})\|Wx\|^2 \leq \|AWx\|^2 \leq (1 + \delta_{2s})\|Wx\|^2, \ \forall \ Wx \in \Sigma_{2s}, \tag{16}$$

when $\tilde{p} = 4C\delta_{2s}^{-2} s \ln(em/s)$ for some constant C. Therefore, when the feature dimension is less than the dimension of input signal, we require in our construction

$$n > 2\tilde{p} = 8C\delta_{2s}^{-2} s \ln(em/s). \tag{17}$$

These lead to the following result:

Theorem 7. *Suppose Ω is s-approximately-sparse with error β and the tuple $(\beta, \delta_{2s}, s, d_0)$ satisfies* (17) *and $\sqrt{(1 + \delta_{2s})/(1 - \delta_{2s})} < (d_0 - 2\beta)/2\beta$ for some $\delta_{2s} \in (0, 1)$. Then, there exists a 2-layer convolutional network $T : \mathbb{R}^{n \times 1} \mapsto \mathbb{R}^{1 \times 2p}$ which is separable with $2p < n$.*

Proof. Let $(w^1, w^2, \ldots, w^J)$ be J kernels and $W : \mathbb{R}^n \mapsto \mathbb{R}^m$ be the associated operator defined as (15) which satisfies $W^\top W = I$. By Theorem 6, there exists $A : \mathbb{R}^{p \times m}$ with $p = 4C\delta_{2s}^{-2} s \ln(em/s)$ such that the inequality (16) holds. Thus, for all $x \in \Omega_0$ and $y \in \Omega_1$, we have

$$\begin{aligned} \|H_s(Wx) - H_s(Wy)\| &\geq \|Wx - Wy\| - \|Wx - H_s(Wx)\| - \|Wy - H_s(Wy)\|, \\ &= \|x - y\| - \|Wx - H_s(Wx)\| - \|Wy - H_s(Wy)\| \geq d_0 - 2\beta. \end{aligned}$$

which implies

$$\begin{aligned} &\|A(Wx) - A(Wy)\| \\ =&\|A(H_s(Wx)) - A(H_s(Wy)) + A(H_{s^c}(Wx)) - A(H_{s^c}(Wy))\| \\ \geq&\sqrt{1 - \delta_{2s}}\|H_s(Wx) - H_s(Wy)\| - \sqrt{1 + \delta_{2s}}(\|H_{s^c}(Wx)\| + \|H_{s^c}(Wy))\|) \\ \geq&(d_0 - 2\beta)\sqrt{1 - \delta_{2s}} - 2\beta\sqrt{1 + \delta_{2s}} > 0, \end{aligned}$$

where $H_{s^c}(x) = x - H_s(x)$.

Let a^i be the i-th row of A, and let $H : \mathbb{R}^{m \times 2} \mapsto \mathbb{R}$ where $\tilde{a}^i = [a^i, -a^i]$. Then

$$H^i(x) = a^i *_V \sigma(Wx) - a^i *_V \sigma(-Wx) = \langle a^i, \sigma(Wx) - \sigma(-Wx) \rangle = \langle a^i, Wx \rangle,$$

since $\sigma(x) - \sigma(-x) = x$. Set the feature map $T : \mathbb{R}^{n\times 1} \mapsto \mathbb{R}^{1\times 2p}$ as

$$T(x) = \left[\sigma \circ H \circ \sigma \circ \begin{bmatrix} W \\ -W \end{bmatrix}(x), \sigma \circ -H \circ \sigma \circ \begin{bmatrix} W \\ -W \end{bmatrix}(x)\right].$$

Thus, $T(x) = [\sigma(AWx), \sigma(-AWx)]$. For all $x \in \Omega_0$ and $y \in \Omega_1$, we have

$$\begin{aligned} \|T(x) - T(y)\|_2^2 &= \|\sigma(AWx) - \sigma(AWy))\|_2^2 + \|\sigma(-AWx)) - \sigma(-AWy))\|_2^2 \\ &\geq \frac{1}{2}\|AWx - AWy\|_2^2 > 0, \end{aligned}$$

where the first inequality is from (14). Therefore, T is separable and $2p = 8C\delta_{2s}^{-2} s \ln(em/s) < n$. □

Remark 2. *Recall that β measures the error incurred by thresholding signals in Ω under the transform W into s-sparse vectors. Thus the condition in Theorem 7 involving β and δ_{2s} represents a trade-off between the approximate sparsity condition and dimension reduction capability. If Ω has better s-sparse approximation, δ_{2s} can be closer to 1 which reduces the required number of measurements, i.e. p can be smaller. Thus, the separable condition becomes easier to satisfy when the gap d_0 between Ω_0 and Ω_1 increases.*

In the above theorem, the support size of the kernel is the same as the input signal. This essentially reduces the convolutional network to a fully connected network, and is seldom used in real world applications. However, we show in the following lemma that can be represented as composition of multiple 1-layer convolutional neural networks and the support of kernels is up to 3 which is common in many practical architectures such as VGG-16 [33], ResNet [14], DenseNet [16], etc. The next lemma shows that each row (equivalently, channel) of A can be decomposed into convolutions with small filters.

Lemma 3. *Let $a \in \mathbb{R}^n$, there exist q_1 filters $\{\alpha_i\}_{i=1}^{q_1}$ with size 3 and q_2 filters $\{\beta_i\}_{i=1}^{q_2}$ with size 2 such that*

$$a = \alpha_1 *_F \ldots \alpha_{q_1} *_F \beta_1 *_F \ldots *_F \beta_{q_2},$$

where $*_F$ denotes the convolution with zero extension and $*_F : \mathbb{R}^n \times \mathbb{R}^r \mapsto \mathbb{R}^{n-r+1}$.

Proof. Let $\mathbf{F}x$ be the Fourier series of the sequence x, i.e.

$$(\mathbf{F}x)(\xi) = \sum_{j=0}^{n-1} \mathbf{x}_n \exp(-ij\xi).$$

Let $z = \exp(-i\xi)$, then $P(z) := (\mathbf{F}a)(z)$ is a polynomial of z of degree n. By the unique factorization theorem, $P(z)$ can be factorized as

$$\begin{aligned} P(z) &= \prod_{p_1=1}^{q_1} (c_{p_1} z^2 + b_{p_1} z + a_{p_1}) \prod_{p_2=1}^{q_2} (b'_{p_2} z + a'_{p_2}) \\ &= \prod_{p_1=1}^{q_1} (\mathbf{F}\alpha_{p_1})(z) \prod_{p_2=1}^{q_2} (\mathbf{F}\beta_{p_2})(z) \\ &= \mathbf{F}(\alpha_1 *_F \ldots \alpha_{q_1} *_F \beta_1 *_F \ldots *_F \beta_{q_2}) \end{aligned}$$

where $\alpha_{p_1} = [a_{p_1}, b_{p_1}, c_{p_1}] \in \mathbb{R}^3$ and $\beta_{p_2} = [a'_{p_2}, b'_{p_2}] \in \mathbb{R}^2$. Taking the inverse Fourier transform, we get the desired result. □

Theorem 8. *Suppose the condition in Theorem 7 holds. There exists a sequence of 1 layer convolutional layers* $\{\mathbf{G}^1_\sigma, \mathbf{G}^2_\sigma, \ldots, \mathbf{G}^{k+1}_\sigma\}$*, such that the feature map* $T = \mathbf{G}^{k+1}_\sigma \circ \mathbf{G}^k_\sigma \cdots \circ \mathbf{G}^1_\sigma : \mathbb{R}^{n\times 1} \mapsto \mathbb{R}^{1\times 2p}$ *is separable.*

Proof. Let W be the transformation in Theorem 7 which can be seen as the convolution layer, $H \in \mathbb{R}^{p\times m}$ be the valid convolution with kernels from the row of A which are given in Theorem 7.Therefore, it suffices to prove that there exists convolutions map T such that

$$T(x) = [\sigma(H(\sigma(W(x)), \sigma(-W(x)))), \sigma(-H(\sigma(W(x)), \sigma(-W(x))))]. \tag{18}$$

Let $\mathbf{G}^1_\sigma = [\sigma(Wx), \sigma(-Wx)] \in \mathbb{R}^{m\times 2}$.

By Lemma 3, each row of A can be decomposed into compositions of up to k short filters with size 3, inserting delta filters when necessary,

$$a^i = g^i_1 *_F g^i_2 \cdots *_F g^i_k, \quad \forall i = 1, \ldots, p.$$

Without loss of generality, assume that $m = 2k + 1$. Otherwise, we can add some zeros entries to the input signal x.
Define $U^i_1 = g^i_1$ and $\widetilde{U}^i_1 = [g^i_1, -g^i_1] \in \mathbb{R}^{3\times 2}$ for each $j = 1, \ldots p$. Let $\mathbf{G}^2_\sigma$ be the 1-layer convolution with kernels $\{\widetilde{U}^i_1\}^p_{i=1} \cup \{-\widetilde{U}^i_1\}^p_{i=1}$. Then, $\mathbf{G}^2_\sigma = (G^{2,1}_\sigma, \ldots, G^{2,2p}_\sigma)$ is a map from $\mathbb{R}^{m\times 2}$ to $\mathbb{R}^{(m-2)\times 2p}$ and

$$\begin{aligned}
G^{2,i}_\sigma \circ \mathbf{G}^1_\sigma &= \sigma((\sigma(Wx) - \sigma(-Wx)) *_V g^i_1) = \sigma(Wx *_V g^i_1), i = 1, \ldots, p,\\
G^{2,i}_\sigma \circ \mathbf{G}^1_\sigma &= \sigma(-(\sigma(Wx) - \sigma(-Wx)) *_V g^{i-p}_1) = \sigma(Wx *_V -g^{i-p}_1), i = p+1, \ldots, 2p.
\end{aligned}$$

Define $U^i_j \in \mathbb{R}^{3\times p}$ to be the zero matrix except the i-th column equals to the $g^i_j(-\cdot)$ for all $i = 1, \ldots, p$ and $j = 2, \ldots, k$, $\widetilde{U}^i_j = [U^i_j, -U^i_j] \in \mathbb{R}^{3\times 2p}$ and $\mathbf{G}^{j+1}_\sigma$ be a 1-layer convolution with kernels $\{\widetilde{U}^i_j\}^p_{i=1} \cup \{-\widetilde{U}^i_j\}^p_{i=1}$. Thus, $\mathbf{G}^{j+1}_\sigma = (G^{j+1,1}_\sigma, \ldots, G^{j+1,2p}_\sigma)$ is a map from $\mathbb{R}^{(m-2j)\times 2p}$ to $\mathbb{R}^{(m-2(j+1))\times 2p}$ and

$$\begin{aligned}
G^{j+1,i}_\sigma \circ \mathbf{G}^j_\sigma \circ \cdots \circ \mathbf{G}^1_\sigma &= \sigma(Wx *_V g^i_1 *_V \cdots *_V g^i_j(-\cdot)), \quad i = 1, \ldots, p,\\
G^{j+1,i}_\sigma \circ \mathbf{G}^j_\sigma \circ \cdots \circ \mathbf{G}^1_\sigma &= \sigma(-Wx * g^{i-p}_1 *_V \cdots *_V g^i_j(-\cdot)), \quad i = p+1, \ldots, 2p.
\end{aligned}$$

Define

$$T = \mathbf{G}^{k+1}_\sigma \circ \mathbf{G}^k_\sigma \cdots \circ \mathbf{G}^1_\sigma : \mathbb{R}^n \mapsto \mathbb{R}^{1\times 2p}.$$

Then, we know feature map T has the form (18). □

We have established the existence of composed convolutional networks with fixed support kernel size that is a separable stable feature extractor. In addition, when the nonlinear activation is ReLU, one can see from the above arguments that there exists composed convolutional network $H_\sigma = \mathbf{G}^k_\sigma \circ \mathbf{G}^{k-1}_\sigma \cdots \circ \mathbf{G}^1_\sigma$ such that

$$H_\sigma(x) = [\sigma(Ax + b), \sigma(-Ax - b)].$$

for some $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$. The above result implies that from the perspective of approximation capability, composed convolutional network is at least as strong as fully connected network because we can actually implement a fully connected network with one hidden layer by a convolutional neural network.

2.5 The advantage of nonlinearity

The previous section basically constructs a convolutional network with fixed filter size to represent a linear operator that has the separable property in feature space. However, this representation does not show the advantage of nonlinear activations in neural networks. We now investigate the effects of the non-linear activation functions in deep neural networks in following simple context.

Define a L-layer deep neural networks to be

$$f(x) = g \circ \sigma(W_L(\dots \sigma(W_1 x + b_1)\dots) + b_L),$$

We will prove that a nonlinear transformation of the inputs may further decrease the approximation error, whereas a linear transformation will not. The result depends on the following assumptions.

Assumption 3. *The minimum of problem*

$$\min_f E(f) = \int_\Omega (f^*(x) - f(x))^2 d\mu \tag{19}$$

is bigger than 0 and the minimizer is attainable.

Assumption 4. *The activation function (applied point-wise) σ satisfies*

1. $\sigma \in C^\infty(\mathbb{R})$
2. *σ is non-degenerate, i.e. for each $k \geq 0$ there is a $b_k \in \mathbb{R}$ such that $\frac{d^k\sigma}{dx^k}(b_k) \neq 0$.*

Rewrite $f(x)$ as $f(x) = \tilde{g}(W_1 x + b_1)$ for some function $\tilde{g}$ and $\tilde{g} \in \widetilde{\mathcal{G}}$. Let $\tilde{f} = \tilde{g}^*(W_1^* x + b_1^*)$ be the minimizer of (19) and $E(\tilde{f}) > 0$. Define the linear transformation $P(x) = x + V(Ux + b)$ for $V \in \mathbb{R}^{n\times q}$, $U \in \mathbb{R}^{q\times p}$, $b \in \mathbb{R}^q$. In the next theorem, we first show that the linear transformation does not decrease the approximation error.

Theorem 9. *The following identity holds:*

$$E(\tilde{f}) = \min_{f,U,V,b} \int_\Omega (f^*(x) - f \circ P(x; V, U, b))^2 d\mu. \tag{20}$$

Proof. Since $f(x) = f \circ P(x; 0, U, b)$ for all $x \in \Omega$, we have

$$E(\tilde{f}) \geq \min_{f,U,V,b} \int_\Omega (f^*(x) - f \circ P(x; V, U, b))^2 d\mu.$$

Moreover, and we have

$$f \circ P(x; V, U, b) = \tilde{g}(W_1(x + V(Ux + b)) + b_1) = \tilde{g}(W_1^{'} x + b_1^{'})$$

where $W_1^{'} = W_1 + VU$ and $b_1^{'} = b_1 + W_1 V b$. Then, we can conclude (20) holds. □

Define the nonlinear transformation $P_\sigma(x; V, U, b) = x + V\sigma(Ux + b)$ and set

$$\mathcal{X} = \{x \in \Omega | \tilde{f}(x) \neq f^*(x)\} \quad \text{and} \quad \mathcal{X}_i = \{x \in \Omega | \nabla_i \tilde{f}(x) \neq 0\},\ i = 1, 2, \ldots, n.$$

We will show that $f \circ P_\sigma$ can decrease the approximation error in the next theorem.

Theorem 10. *Suppose the Assumption 3, 4 hold and $\mu(\mathcal{X} \cap (\cup_{i=1}^n \mathcal{X}_i)) > 0$. Then, there exists some $\tilde{V}$, $\tilde{U}$ and $\tilde{b}$ such that $E(\tilde{f}) > E(\tilde{f} \circ P_\sigma(x; \tilde{V}, \tilde{U}, \tilde{b}))$.*

Proof. We first show that there exist $\tilde{U}$ and $\tilde{b}$ such that

$$\nabla_V E(\tilde{f} \circ P_\sigma(x; 0, \tilde{U}, \tilde{b})) \neq 0. \tag{21}$$

We prove (21) by contradiction. By the dominated convergence theorem, we have

$$\nabla_V E(\tilde{f} \circ P_\sigma(x; 0, U, b)) = -2 \int_\Omega (\tilde{f}(x) - f^*(x)) \nabla \tilde{f}(x) [\sigma(Ux + b)]^\top d\mu = 0, \forall U, b. \tag{22}$$

Since $\mu(\mathcal{X} \cap (\cup_{i=1}^n \mathcal{X}_i)) > 0$, there exists some i_0 such that $\mu(\mathcal{X} \cap \mathcal{X}_{i_0}) > 0$. From (22), we have

$$\int_\Omega (\tilde{f}(x) - f^*(x)) \nabla_{i_0} \tilde{f}(x) [\sigma(u^\top x + b)] d\mu = 0,\ \forall u \in \mathbb{R}^n, b \in \mathbb{R}.$$

For each $k \geq 0$ and multi-index $i = (i_1, \ldots, i_n)$ with $i_j \geq 0$ and $\sum_j i_j = k$,

$$0 = \frac{\partial^k}{\Pi_j \partial u_j^{i_j}} \int_\Omega (\tilde{f}(x) - f^*(x)) \nabla_{i_0} \tilde{f}(x) \sigma(u^\top x + b) d\mu|_{u=0, b=b_k}$$

By assumption 4, for each k we may pick $b_k \in \mathbb{R}$ such that $d^k \sigma(b_k)/dx^k \neq 0$. The above equality then implies

$$0 = \int_\Omega (\tilde{f}(x) - f^*(x)) \nabla_{i_0} \tilde{f}(x) \Pi_j (x_j)^{i_j} d\mu.$$

In particular, $(\tilde{f} - f^*) \nabla_{i_0} \tilde{f}$ is orthogonal to every monomial, and hence must equal 0 a.e. on Ω, which contradicts the assumption that $\mu(\mathcal{X} \cap \mathcal{X}_{i_0}) > 0$. This proves (21).

Next, Define $\hat{V}(\alpha) = -\alpha \nabla_V E(\tilde{f} \circ P_\sigma(x; 0, \tilde{U}, \tilde{b}))$ for $\alpha \in (0, 1)$. By the Taylor's theorem, we have

$$\begin{aligned} E(\tilde{f} \circ P_\sigma(x; \hat{V}(\alpha), \tilde{U}, \tilde{b})) \leq & E(\tilde{f} \circ P_\sigma(x; 0, \tilde{U}, \tilde{b})) - \alpha C^2 \\ & + \frac{1}{2} \alpha^2 C^2 \max_{\|V\| \leq C} \|\nabla_V^2 E(\tilde{f} \circ P_\sigma(x; V, \tilde{U}, \tilde{b}))\|_2 \end{aligned}$$

where $C := \|\nabla_V E(\tilde{f} \circ P_\sigma(x; 0, \tilde{U}, \tilde{b}))\| > 0$. Since E is twice continuously differentiable in V, the last term is finite and $O(\alpha^2)$, hence for sufficiently small α, we have

$$E(\tilde{f} \circ P_\sigma(x; \hat{V}(\alpha), \tilde{U}, \tilde{b})) < E(\tilde{f} \circ P_\sigma(x; 0, \tilde{U}, \tilde{b})) = E(\tilde{f}),$$

which completes the proof. □

Remark 3. *Note that the non-degeneracy condition in assumption 4 was crucial in the proof above to show that there exists a non-zero derivative of the error E. Linear and higher order polynomial activations do not satisfy the degeneracy condition, and hence we cannot ensure improvement using the argument above.*

Remark 4. *We now discuss the condition $\mu(\mathcal{X} \cap (\cup_{i=1}^{n} \mathcal{X}_i)) > 0$. First, since $E(\tilde{f}) > 0$, we must have $\mu(\mathcal{X}) > 0$. In order to have an improvement via gradient descent, in $\mathcal{X}$ we must have some input coordinates that is not linked entirely to "dead" neurons, i.e. we must be able to affect the output of $\tilde{f}$ by changing our inputs by a small amount. The condition $\mu(\mathcal{X} \cap (\cup_{i=1}^{n} \mathcal{X}_i)) > 0$ says precisely this.*

3 Scaling Analysis of Convolutional networks

In this section, we consider the advantages of hierarchical structures used in CNNs by scaling analysis. Assume the oracle function $f^*(x) : \Omega = [0,1]^n \mapsto [0,1]$, i.e. the oracle has bounded range. Moreover, we assume that the oracle function o belongs to the compositional function space defined as follows.

Definition 4. *The compositional function space $C^c(\Omega)$ where $n = rp$ is that for any $F \in C^c(\mathbb{R}^n)$ the function F has the compositional form: $F = H \circ \mathbf{G}$, $\mathbf{G} = (G^1, G^2, \ldots, G^p)$, $G^i = g^i \circ \mathcal{T}^i, g : \mathbb{R}^r \mapsto \mathbb{R}$, $\mathcal{T}^i : \Omega \mapsto \mathbb{R}^r$ where $\mathcal{T}^i(x) = (x_{(i-1)*r+1}, \ldots, x_{i*r})$ for all i, $H : \mathbb{R}^p \mapsto \mathbb{R}$.*

The form defined in 4 is the simplest one with a compositional hierarchical structure. And many more complex hierarchical functions can be constructed from compositions and linear combinations of this form. Note that the assumption that all G^is are the same function can be easily generalized to different functions so long as the dimensions match.

Since the target oracles are usually obtained from the human labels, it is natural to assume that f^* is stable with respect to small perturbations η. Let $\eta = \frac{\nabla f^*(x)}{\|\nabla f^*(x)\|}\sqrt{n}\epsilon$, i.e. the perturbation is on average $O(\epsilon)$ for each dimension. In practice, the oracle should be stable to this kind of perturbation. That is,

$$|f^*(x+\eta) - f^*(x)| \leq \sqrt{n}\|\nabla f^*(x)\|\epsilon + O(\|\epsilon\|^2) \ll 1$$

So we must have $\|\nabla f^*(x)\| = O(1/\sqrt{n})$. Otherwise, the oracle is not stable and is vulnerable to the well known adversarial perturbations [34, 13]. Therefore, it is reasonable to make the following assumption:

Assumption 5. *The oracle o belongs to $C^c(\Omega)$ and $L(H) = O(1/\sqrt{p})$, $L(g) = O(1/\sqrt{r})$, where $L(H)$ and $L(g)$ are the Lipschitz constants for H and g, respectively.*

The above assumption implies the Lipschitz constant of o is $O(1/\sqrt{n})$.

Definition 5. *Let $N(\varepsilon, n; f)$ to be the minimum number of hidden units needed to achieve ε approximation accuracy of f using fully connected networks with one hidden layer.*

Definition 6. *The $(\varepsilon, \alpha, \beta, n)$-class function space $\mathcal{H} \subset C(\mathbb{R}^n)$ such that $N(\varepsilon, n; f) = O(n^\alpha/\varepsilon^\beta)$ for all $f \in \mathcal{H}$.*

Regarding the number of total parameters needed to achieve a desired accuracy, we have the following estimate.

Theorem 11. *Suppose Assumption 5 holds. If o belongs to $(\varepsilon, \alpha, \beta, n)$-class, g belongs to $(\varepsilon, \alpha, \beta, r)$-class and H belongs to $(\varepsilon, \alpha, \beta, p)$-class. Let S_1 be the minimal total number of parameters needed to achieve ε accuracy for the fully connected network. Then,*

$$S_1 = O\left(\frac{n^{\alpha+1}}{\varepsilon^\beta}\right).$$

Moreover, there is a convolutional network $\hat{\mathbf{G}}$ and a fully connected network $\hat{H}$ such that

$$\|\hat{H} \circ \hat{\mathbf{G}} - o\| \leq \varepsilon \quad \text{and} \quad S_2 = O\left(\frac{r^{\alpha+1}}{\varepsilon^\beta} + \frac{p^{\alpha+1}}{\varepsilon^\beta}\right),$$

where S_2 is the total number of parameters of $\hat{H}$ and $\hat{\mathbf{G}}$.

Proof. Up to leading order,

$$S_1 = N(\varepsilon, n; o)(n+2) = O\left(\frac{n^\alpha}{\varepsilon^\beta}\right)(n+2) = O\left(\frac{n^{\alpha+1}}{\varepsilon^\beta}\right).$$

Since g belongs to $(\varepsilon, \alpha, \beta, r)$-class, there exists $q_g = O(r^\alpha/\varepsilon^\beta)$ hidden units such that

$$\|g - \tilde{g}\| \leq \varepsilon/2D_H,$$

where $\tilde{g} = \sum_{i=1}^{q_g} a_i^g \sigma(w_i^{g\top} x + b_i^g)$ and $D_H/\sqrt{n}$ is the Lipschitz constant of H and D_H is a constant. Similarly, since H belongs to $(\varepsilon, \alpha, \beta, p)$-class, there exists $q_H = O(p^\alpha/\varepsilon^\beta)$ hidden units such that

$$\|H - \hat{H}\| \leq \varepsilon/2,$$

where $\hat{H} = \sum_{i=1}^{q_H} a_i^H \sigma(w_i^{H\top} x + b_i^H)$.

Define a convolutional network $\hat{\mathbf{G}} : \mathbb{R}^d \mapsto \mathbb{R}^{p\times 1}$ as follows

$$\mathbb{R}^{rm\times 1} \to \mathrm{conv}_{r,q_g,r} \to \sigma \to \mathrm{conv}_{1,1,0} \to \mathbb{R}^{m\times 1}$$

where $\mathrm{conv}_{k,c,s}$ represents the convolution layer with kernel size k, number of channels c and stride s. The set of kernels in the first layer are $\{w_i^g\}_{i=1}^{q_g}$ and the 1×1 convolution in the last layer is $\{a_i^g\}_{i=1}^{q_g}$. Then, we know $\hat{\mathbf{G}}(x) = (G^1, G^2, \ldots, G^p)$ and $G^i = \hat{g} \circ \mathcal{T}_i$ for all $i = 1, \ldots, p$.

Therefore,

$$\begin{aligned}
\|H(\mathbf{G}) - \hat{H}(\hat{\mathbf{G}})\|_2 &\leq \|H(\mathbf{G}) - H(\hat{\mathbf{G}})\|_2 + \|H(\hat{\mathbf{G}}) - \hat{H}(\hat{\mathbf{G}})\|_2 \\
&= \left(\int_\Omega |H(\mathbf{G}(x)) - H(\hat{\mathbf{G}}(x))|^2 d\mu(x)\right)^{1/2} + \left(\int_\Omega |H(\hat{\mathbf{G}}(x)) - \hat{H}(\hat{\mathbf{G}}(x))|^2 d\mu(x)\right)^{1/2} \\
&\leq \left(\int_\Omega D_H^2 p^{-1} \|\mathbf{G}(x) - \hat{\mathbf{G}}(x)\|_2^2 d\mu(x)\right)^{1/2} + \left(\int_\Omega |H(\hat{\mathbf{G}}(x)) - H(\hat{\mathbf{G}}(x))|^2 d\mu(x)\right)^{1/2} \\
&\leq \left(\int_\Omega D_H^2 p^{-1} \sum_{i=1}^{p} \frac{\varepsilon^2}{4D_H^2} d\mu(x)\right)^{1/2} + \left(\int_\Omega |H(\hat{\mathbf{G}}(x)) - \hat{H}(\hat{\mathbf{G}}(x))|^2 d\mu(x)\right)^{1/2} \\
&\leq \frac{1}{2}\varepsilon + \left(\int_\Omega \frac{1}{4}\varepsilon^2 d\mu(x)\right)^{1/2} = \varepsilon,
\end{aligned}$$

where the second inequality is due to the Lipschitz continuity of H. Moreover, the number of parameters in $\hat{H} \circ \hat{\mathbf{G}}$ is

$$S_2 = N\left(\frac{\varepsilon}{2D_H}, r; g\right) r + N\left(\frac{\varepsilon}{2}, p; H\right) p = O\left(\frac{r^{\alpha+1}}{\varepsilon^\beta} + \frac{p^{\alpha+1}}{\varepsilon^\beta}\right),$$

which completes the proof. □

Although the exact approximation rate of a fully connected network with one hidden layer is difficult, the upper bound of such estimation exists in several cases. For example, the function class in [2] is approximated with rate $N(\varepsilon, n; f) = O(C_f^2/\varepsilon)$, where $C_f = \int \|\omega\||\hat{f}(\omega)|d\omega$ depends polynomially or exponentially on d. In [1], some interesting cases with $C_f = O(n)$ are given, i.e. $\alpha = 2, \beta = 1$. From the above theorem, we see the obvious advantage of composed convolutional network. The bounds constitute as sufficient conditions.

Moreover, the above argument could be easily generalized to convolutional networks with multiple hidden layers as shown in next corollary.

Corollary 1. *Let $F : \Omega \mapsto \mathbb{R}$, $n = r^L$, $g_1, \ldots, g_L : \mathbb{R}^r \mapsto \mathbb{R}$. $F = g_L \circ \mathbf{G}^L$, $\mathbf{G}^l : \mathbb{R}^{r^{L-l+1}} \mapsto \mathbb{R}^r, \mathbf{G}^l = (G^{l,1}, G^{l,2}, \ldots, G^{l,r})$, $G^{l,i} : \mathbb{R}^{r^{L-l+1}} \mapsto \mathbb{R}$. $G^l = g^l \circ \mathbf{G}^{l-1}, l = 1, \ldots, L$. Assume $n \gg r, Lip(g_l) = O(1/\sqrt{r}), \forall\, l = 1, \ldots, L$. If F belongs to $(\varepsilon, \alpha, \beta, n)$-class, g belongs to $(\varepsilon, \alpha, \beta, r)$-class, then the total number of parameters S_1, of using fully connected network to approximate F, is*

$$S_1 = O\left(\frac{n^{\alpha+1}}{\varepsilon^\beta}\right),$$

and there exists a convolutional network that achieves ε-accuracy with total number of parameters

$$S_2 = O\left(\frac{r^{\alpha+1}}{\varepsilon^\beta}(\log n)^{\beta+1}\right).$$

Proof. Similar to the proof of Theorem 11, we only need to approximate each $g_l : \Omega \mapsto \mathbb{R}$ to $O(\frac{\varepsilon}{L})$, then entire approximation will be in order of $O(\varepsilon)$. So

$$\begin{aligned} S_2 &= N\left(\frac{\varepsilon}{L}, r\right) r + N\left(\frac{\varepsilon}{L}, r\right) r + \cdots N\left(\frac{\varepsilon}{L}, r\right) r \\ &= LN\left(\frac{\varepsilon}{L}, r\right) r = O\left(\frac{r^{\alpha+1}}{\varepsilon^{\beta}} L^{\beta+1}\right), \end{aligned}$$

which completes the proof. □

Thus, using the CNN architecture, the number of parameter used only depends on $(\log n)^{\beta+1}$, which is much better than $O(n^{\alpha+1})$ of fully-connected networks. In the usual case like image recognition where n is large, this means that for hierarchical functions, using "deep" convolutional networks is exponentially more parameter efficient than fully connected network. Theorem 11 and Corollary 1 establish that using convolutional networks to approximate compositional functions has huge advantages. Moreover, observe that sampling errors can also be shown to be improved using a similar analysis. For example, define $N(m, \varepsilon)$ to be the number of samples needed to achieve ε accuracy for a fully connected network with one hidden layer and S parameters. Assume $N(S, \varepsilon) = O(\frac{m^{\alpha}}{\varepsilon^{\beta}})$, then similar scaling arguments can be used to characterize the sampling error. We see from the above proof that when comparing network structures, the result is independent of β. This implies that composed convolutional network also has much better sample error estimates.

4 Conclusion

In this paper, we proved that under suitable conditions, convolution neural networks can inherit the universal approximation property of its last fully connected layers. Moreover, we show that nonlinearity in the transformations is important to allow for local improvements of the approximation via composition. Finally, we proved that when the target function class has a compositional structure, using convolutional networks requires fewer parameters, less computation and fewer samples compared with fully connected networks achieving the same accuracy. When the compositional structure is hierarchical, the reduction in parameters is exponential in the number of compositional levels.

An interesting future direction is to explore the class of compositional hierarchical functions. Many functions resulted from Markovian physical process are of this form, it would be interesting to model such processes and develop a generative mechanism for such functions.

Appendix A Proof of theorem 3

The next Lemma shows the bound of two functions f_1, f_2 in $\mathcal{H}^{L,K}$ under $\mathbb{E}$ and $\mathbb{E}_z$.

Lemma 4. *For any $f_1, f_2 \in \mathcal{H}(\Omega)$, the following inequalities hold:*

$$\begin{aligned} |\mathbb{E}_z(f_1) - \mathbb{E}_z(f_2)| &\leq 2M\|f_1 - f_2\|_\infty \\ |\mathbb{E}(f_1) - \mathbb{E}(f_2)| &\leq 2M\|f_1 - f_2\|_\infty. \end{aligned} \tag{23}$$

Proof. Since

$$\begin{aligned} &\sum_{i=1}^{m}(f_1(x_i) - o(x_i))^2 - \sum_{i=1}^{m}(f_2(x_i) - o(x_i))^2 \\ =&\sum_{i=1}^{m}(f_1(x_i) - f_2(x_i))(f_1(x_i) - o(x_i) + f_2(x_i) - o(x_i)) \\ \leq&2mM\|f_1 - f_2\|_\infty. \end{aligned}$$

Then the first inequality in (23) holds. Similarly, we also have

$$\begin{aligned} &|\mathbb{E}(f_1) - \mathbb{E}(f_2)| \\ \leq&\int_\Omega |(f_1(x) - f_2(x))(f_1(x) - c(x) + f_2(x) - c(x))|d\mu(x) \\ \leq&2M\|f_1 - f_2\|_\infty. \end{aligned}$$

This ends the proof. □

Let $\{f_j\}_{j=1}^k$, where $K = \mathcal{N}_{\mathcal{H}_{L,K}}(\frac{\gamma\delta}{2})$ is a sequence such that $\mathcal{H}_{L,K}$ is covered by L^∞ balls in $\mathcal{H}^{L,K}$ centered at f_j with radius $\frac{\gamma\delta}{2}$. Define $\Omega_j = \{f : \|f - f_j\| \leq \frac{\gamma\delta}{2M}\}$, $1 \leq j \leq k$. Then we have $\mathcal{H}^{L,K} \subset \cup_{j=1}^k \Omega_j$. For each j, denote $\xi = (f_j(\zeta) - o(\zeta))^2$ as a random variable, where ζ is a i.i.d random variable drawn from the distribution $\rho = \mu$ on Ω. Note that $f_j \in \mathcal{H}^{L,K}$ and the definition of c, implies that $\|f_j\|_\infty \leq 1$ and $\|c\|_\infty \leq 1$. Thus we have $\xi - E\xi \leq 1$. Furthermore,

$$E\xi^2 = E(f_j(\zeta) - c(\zeta))^4 \leq E(f_j(\zeta) - c(\zeta))^2 = E\xi.$$

Applying Lemma 1 to ξ with $B = \eta = 1$, we have

$$\mathbb{P}\left\{\frac{\mathbb{E}(f_j) - \mathbb{E}_z(f_j)}{\sqrt{\mathbb{E}(f_j) + \delta}} > \gamma\sqrt{\delta}\right\} \leq \exp\left\{-\frac{3\gamma^2 m\delta}{8}\right\}$$

where we used the fact that

$$E\xi = E(f(\zeta) - c(\zeta))^2 = \int_\Omega (f(x) - c(x))^2 d\mu(x) = \mathbb{E}(f).$$

By the definition of covering number, for any function $f \in \mathcal{H}^{L,K}$, there exists a function f_j such that $\|f - f_j\| \leq \frac{\gamma\delta}{2M}$. This together with Lemma 4, yields

$$|\mathbb{E}_z(f) - \mathbb{E}_z(f_j)| \leq 2M\|f - f_j\|_\infty \leq \gamma\delta,$$
$$|\mathbb{E}(f) - \mathbb{E}(f_j)| \leq 2M\|f - f_j\|_\infty \leq \gamma\delta.$$

Therefore, since $\mathbb{E}(f) \geq 0$,

$$\frac{|\mathbb{E}_z(f) - \mathbb{E}_z(f_j)|}{\sqrt{\mathbb{E}(f) + \delta}} \leq \gamma\sqrt{\delta}, \quad \frac{|\mathbb{E}(f) - \mathbb{E}(f_j)|}{\sqrt{\mathbb{E}(f) + \delta}} \leq \gamma\sqrt{\delta}.$$

By the second inequality, we have

$$\begin{aligned}\mathbb{E}(f_j) + \delta &= \mathbb{E}(f_j) - \mathbb{E}(f) + \mathbb{E}(f) + \delta \\ &\leq \gamma\sqrt{\delta}\sqrt{\mathbb{E}(f) + \delta} + \mathbb{E}(f) + \delta \\ &\leq \sqrt{\delta}\sqrt{\mathbb{E}(f) + \delta} + \mathbb{E}(f) + \delta \leq 2(\mathbb{E}(f) + \delta),\end{aligned}$$

which leads to $\sqrt{\mathbb{E}(f_j) + \delta} \leq 2\sqrt{\mathbb{E}(f) + \delta}$ for any f_j.

Now, if we assume that $\frac{\mathbb{E}(f) - \mathbb{E}_z(f)}{\sqrt{\mathbb{E}(f)+\delta}} > 4\gamma\sqrt{\delta}$, then we have

$$\begin{aligned}\frac{\mathbb{E}(f_j) - \mathbb{E}_z(f_j)}{2\sqrt{\mathbb{E}(f) + \delta}} \geq& \frac{\mathbb{E}(f) - \mathbb{E}_z(f)}{2\sqrt{\mathbb{E}(f) + \delta}} - \frac{\mathbb{E}(f) - \mathbb{E}(f_j)}{2\sqrt{\mathbb{E}(f) + \delta}} - \frac{\mathbb{E}_z(f_j) - \mathbb{E}_z(f)}{2\sqrt{\mathbb{E}(f) + \delta}} \\ >& 2\gamma\sqrt{\delta} - \frac{\gamma\sqrt{\delta}}{2} - \frac{\gamma\sqrt{\delta}}{2} = \gamma\sqrt{\delta}.\end{aligned}$$

Since we have $\sqrt{\mathbb{E}(f_j) + \delta} \leq 2\sqrt{\mathbb{E}(f) + \delta}$ for any $f \in \Omega_j$, then if the condition $\frac{\mathbb{E}(f) - \mathbb{E}_z(f)}{\sqrt{\mathbb{E}(f)+\delta}} > 4\gamma\sqrt{\delta}$ holds, then the following inequality

$$\frac{\mathbb{E}(f_j) - \mathbb{E}_z(f_j)}{2\sqrt{\mathbb{E}(f_j) + \delta}} \geq \frac{\mathbb{E}(f_j) - \mathbb{E}_z(f_j)}{2\sqrt{\mathbb{E}(f) + \delta}} > \gamma\sqrt{\delta}$$

holds. Hence, for each fixed j, $1 \leq j \leq k$,

$$\mathbb{P}\left\{\sup_{f\in\Omega_j} \frac{\mathbb{E}(f) - \mathbb{E}_z(f)}{\sqrt{\mathbb{E}(f) + \delta}} > 4\gamma\sqrt{\delta}\right\} \leq \mathbb{P}\left\{\frac{\mathbb{E}(f_j) - \mathbb{E}_z(f_j)}{\sqrt{\mathbb{E}(f_j) + \delta}} > \gamma\sqrt{\delta}\right\}.$$

Since $\mathcal{H}^{L,K} \subset \cup_{j=1}^k \Omega_j$, we have

$$\mathbb{P}\left\{\sup_{f\in\mathcal{H}(\Omega)} \frac{\mathbb{E}(f) - \mathbb{E}_z(f)}{\sqrt{\mathbb{E}(f) + \delta}} > 4\gamma\sqrt{\delta}\right\} \leq \sum_{j=1}^k \mathbb{P}\left\{\sup_{f\in\Omega_j} \frac{\mathbb{E}(f) - \mathbb{E}_z(f)}{\sqrt{\mathbb{E}(f) + \delta}} > 4\gamma\sqrt{\delta}\right\},$$

combined with (24), (25) and $k = \mathcal{N}_{\mathcal{H}^{L,K}}(\frac{\gamma\delta}{2})$, yields

$$\mathbb{P}\left\{\sup_{f\in\mathcal{H}^{L,K}} \frac{\mathbb{E}(f) - \mathbb{E}_z(f)}{\sqrt{\mathbb{E}(f) + \delta}} > 4\gamma\sqrt{\delta}\right\} \leq \mathcal{N}_{\mathcal{H}^{L,K}}\left(\frac{\gamma\delta}{2M}\right) \exp\left\{-\frac{3\gamma^2 m\delta}{8}\right\}.$$

This completes the proof.

References

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

[2] Barron, A.R.: Approximation and estimation bounds for artificial neural networks. Machine Learning **14**(1), 115–133 (1994)

[3] Cai, J.F., Ji, H., Shen, Z., Ye, G.B.: Data-driven tight frame construction and image denoising. Applied and Computational Harmonic Analysis **37**(1), 89–105 (2014)

[4] Cai, J.F., Shen, Z., Ye, G.B.: Approximation of frame based missing data recovery. Applied and Computational Harmonic Analysis **31**(2), 185–204 (2011)

[5] Candes, E.J.: The restricted isometry property and its implications for compressed sensing. Comptes Rendus Mathematique **346**(9-10), 589–592 (2008)

[6] Candes, E.J., Tao, T.: Decoding by linear programming. IEEE transactions on information theory **51**(12), 4203–4215 (2005)

[7] Cohen, N., Sharir, O., Shashua, A.: On the expressive power of deep learning: A tensor analysis. In: Conference on Learning Theory, pp. 698–728 (2016)

[8] Cohen, N., Shashua, A.: Convolutional rectifier networks as generalized tensor decompositions. In: International Conference on Machine Learning (ICML) (2016)

[9] Cucker, F., Smale, S.: On the mathematical foundations of learning. Bulletin of the American mathematical society **39**(1), 1–49 (2002)

[10] Cybenko, G.: Approximation by superpositions of a sigmoidal function. Mathematics of control, signals and systems **2**(4), 303–314 (1989)

[11] Foucart, S., Rauhut, H.: A mathematical introduction to compressive sensing, vol. 1. Birkhäuser Basel (2013)

[12] Giryes, R., Sapiro, G., Bronstein, A.M.: Deep neural networks with random gaussian weights: A universal classification strategy? IEEE Trans. Signal Processing **64**(13), 3444–3457 (2016)

[13] Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014)

[14] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR, pp. 770–778 (2016)

[15] Hornik, K.: Approximation capabilities of multilayer feedforward networks. Neural networks **4**(2), 251–257 (1991)

[16] Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: CVPR, vol. 1, p. 3 (2017)

[17] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp. 1097–1105 (2012)

[18] Le Roux, N., Bengio, Y.: Representational power of restricted boltzmann machines and deep belief networks. Neural computation **20**(6), 1631–1649 (2008)

[19] Le Roux, N., Bengio, Y.: Deep belief networks are compact universal approximators. Neural computation **22**(8), 2192–2207 (2010)

[20] Leshno, M., Lin, V.Y., Pinkus, A., Schocken, S.: Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. Neural networks **6**(6), 861–867 (1993)

[21] Lin, H., Jegelka, S.: Resnet with one-neuron hidden layers is a universal approximator. In: Advances in Neural Information Processing Systems, pp. 6170–6179 (2018)

[22] Lu, Z., Pu, H., Wang, F., Hu, Z., Wang, L.: The expressive power of neural networks: A view from the width. In: Advances in Neural Information Processing Systems, pp. 6231–6239 (2017)

[23] Mallat, S.: Group invariant scattering. Communications on Pure and Applied Mathematics **65**(10), 1331–1398 (2012)

[24] Mallat, S.: Understanding deep convolutional networks. Phil. Trans. R. Soc. A **374**(2065), 20150203 (2016)

[25] Mhaskar, H., Liao, Q., Poggio, T.: Learning functions: when is deep better than shallow. arXiv preprint arXiv:1603.00988 (2016)

[26] Mhaskar, H.N., Poggio, T.: Deep vs. shallow networks: An approximation theory perspective. Analysis and Applications **14**(06), 829–848 (2016)

[27] Montufar, G., Ay, N.: Refinements of universal approximation results for deep belief networks and restricted boltzmann machines. Neural Computation **23**(5), 1306–1319 (2011)

[28] Montufar, G.F., Pascanu, R., Cho, K., Bengio, Y.: On the number of linear regions of deep neural networks. In: Advances in neural information processing systems, pp. 2924–2932 (2014)

[29] Papyan, V., Romano, Y., Elad, M.: Convolutional neural networks analyzed via convolutional sparse coding. The Journal of Machine Learning Research **18**(1), 2887–2938 (2017)

[30] Pascanu, R., Montufar, G., Bengio, Y.: On the number of response regions of deep feed forward networks with piece-wise linear activations. arXiv preprint arXiv:1312.6098 (2013)

[31] Poggio, T., Shelton, C.R.: On the mathematical foundations of learning. American Mathematical Society **39**(1), 1–49 (2002)

[32] Shaham, U., Cloninger, A., Coifman, R.R.: Provable approximation properties for deep neural networks. Applied and Computational Harmonic Analysis (2016)

[33] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

[34] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013)

[35] Tai, C.: Multiscale adaptive representation of signals: I. the basic framework. Journal of Machine Learning Research **17**(140), 1–38 (2016)

[36] Telgarsky, M.: Benefits of depth in neural networks. arXiv preprint arXiv:1602.04485 (2016)

[37] Vapnik, V.N., Vapnik, V.: Statistical learning theory, vol. 1. Wiley New York (1998)

[38] Wiatowski, T., Bölcskei, H.: A mathematical theory of deep convolutional neural networks for feature extraction. IEEE Transactions on Information Theory **64**(3), 1845–1866 (2018)

[39] Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning requires rethinking generalization. arXiv preprint arXiv:1611.03530 (2016)

[40] Zhou, D.X.: Universality of deep convolutional neural networks. arXiv preprint arXiv:1805.10769 (2018)