Appl. Comput. Harmon. Anal. 48 (2020) 787–794

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Letter to the Editor

Universality of deep convolutional neural networks

Ding-Xuan Zhou

School of Data Science and Department of Mathematics, City University of Hong Kong, Kowloon, Hong Kong

ARTICLE INFO

Article history:
Received 1 August 2018
Received in revised form 5 May 2019
Accepted 10 June 2019
Available online 13 June 2019
Communicated by Charles K. Chui

Keywords:
Deep learning
Convolutional neural network
Universality
Approximation theory

ABSTRACT

Deep learning has been widely applied and brought breakthroughs in speech recognition, computer vision, and many other domains. Deep neural network architectures and computational issues have been well studied in machine learning. But there lacks a theoretical foundation for understanding the approximation or generalization ability of deep learning methods generated by the network architectures such as deep convolutional neural networks. Here we show that a deep convolutional neural network (CNN) is universal, meaning that it can be used to approximate any continuous function to an arbitrary accuracy when the depth of the neural network is large enough. This answers an open question in learning theory. Our quantitative estimate, given tightly in terms of the number of free parameters to be computed, verifies the efficiency of deep CNNs in dealing with large dimensional data. Our study also demonstrates the role of convolutions in deep CNNs.

1. Introduction and main results

Deep learning provides various models and algorithms to process data as efficiently as biological nervous systems or neuronal responses in the human brain [16,11,14,9,15]. It is based on deep neural network architectures and those structures bring essential tools for obtaining data features and function representations in practical applications. A main concern about deep learning which has attracted much scientific attention and some criticism is its lack of theories supporting its practical efficiency caused by its network structures, though there have been some theoretical attempts from approximation theory viewpoints [3,19,21,26]. In particular, for deep CNNs having convolutional structures without fully connected layers, it is unknown which kinds of functions can be approximated. This paper provides a rigorous mathematical theory to answer this question and to illustrate the role of convolutions.

The deep CNNs considered in this paper have two essential ingredients: a rectified linear unit (ReLU) defined as a univariate nonlinear function σ given by

E-mail address: mazhou@cityu.edu.hk.

https://doi.org/10.1016/j.acha.2019.06.004
1063-5203/

$$\sigma(u) = (u)_+ = \max\{u, 0\}, \qquad u \in \mathbb{R} \tag{1.1}$$

and a sequence of convolutional filter masks $\mathbf{w} = \{w^{(j)}\}_j$ inducing sparse convolutional structures. Here a filter mask $w = (w_k)_{k=-\infty}^{\infty}$ means a sequence of filter coefficients. We use a fixed integer filter length $s \geq 2$ to control the sparsity, and assume that $w_k^{(j)} \neq 0$ only for $0 \leq k \leq s$. The convolution of such a filter mask w with another sequence $v = (v_0, \ldots, v_{D-1})$ is a sequence $w*v$ given by $(w*v)_i = \sum_{k=0}^{D-1} w_{i-k} v_k$. This leads to a $(D+s) \times D$ Toeplitz type convolutional matrix T which has constant diagonals:

$$T = \begin{bmatrix} w_0 & 0 & 0 & 0 & \cdots & 0 \\ w_1 & w_0 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ w_s & w_{s-1} & \cdots & w_0 & 0 \cdots & 0 \\ 0 & w_s & \cdots & w_1 & w_0 \cdots & 0 \\ \vdots & \ddots & \ddots \ddots & \ddots & \ddots \ddots & \vdots \\ \cdots & \cdots & 0 & w_s & \cdots & w_0 \\ \cdots & \cdots & \cdots & 0 \quad w_s & \cdots & w_1 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots\cdots & \cdots 0 & w_s & w_{s-1} \\ 0 & \cdots & \cdots & \cdots & \cdots 0 & w_s \end{bmatrix}. \tag{1.2}$$

Sparse matrices of this form induce deep CNNs which are essentially different from the classical neural networks involving full connection matrices. Note that the number of rows of T is s greater than that of columns. This leads us to take a sequence of linearly increasing widths $\{d_j = d + js\}$ for the network, which enables the deep CNN to represent functions of richer structures.

Definition 1. Starting with the action of $T^{(1)}$ on the input data vector $x \in \mathbb{R}^d$, we can define a deep CNN of depth J as a sequence of J vectors $h^{(j)}(x)$ of functions on $\mathbb{R}^d$ given iteratively by $h^{(0)}(x) = x$ and

$$h^{(j)}(x) = \sigma\left(T^{(j)} h^{(j-1)}(x) - b^{(j)}\right), \qquad j = 1, 2, \ldots, J, \tag{1.3}$$

where $T^{(j)} = \left(w_{i-k}^{(j)}\right)$ is a $d_j \times d_{j-1}$ convolutional matrix, σ acts on vectors componentwise, and $\mathbf{b}$ is a sequence of bias vectors $b^{(j)}$.

Except the last iteration, we take $b^{(j)}$ of the form

$$[b_1 \ \ldots b_s \ b_{s+1} \ b_{s+1} \ \ldots \ b_{s+1} \ b_{d_j - s+1} \ \ldots b_{d_j}]^T \tag{1.4}$$

with the $d_j - 2s$ repeated components in the middle. The sparsity of $T^{(j)}$ and the special form of $b^{(j)}$ tell us that the j-th iteration of the deep CNN involves $3s + 2$ free parameters. So in addition to the $2d_J + s + 1$ free parameters for $b^{(J)}, c \in \mathbb{R}^{d_J}, w^{(J)}$, the total number of free parameters in the deep CNN is $(5s+2)J + 2d - 2s - 1$, much smaller than that in a classical fully connected multi-layer neural network with full connection matrices $T^{(j)}$ involving $d_j d_{j-1}$ free parameters. It demonstrates the computational efficiency of deep CNNs.

The hypothesis space of a learning algorithm is the set of all possible functions that can be represented or produced by the algorithm.

Definition 2. For the deep CNN of depth J considered here, the hypothesis space is a set of functions defined by

$$\mathcal{H}_J^{\mathbf{w},\mathbf{b}} = \left\{ \sum_{k=1}^{d_J} c_k h_k^{(J)}(x) : c \in \mathbb{R}^{d_J} \right\}. \tag{1.5}$$

This hypothesis space and its approximation ability depend completely on the sequence of convolutional filter masks $\mathbf{w} = \{w^{(j)}\}_{j=1}^J$ and the sequence of bias vectors $\mathbf{b} = \{b^{(j)}\}_{j=1}^J$. Observe that each function in the hypothesis space $\mathcal{H}_J^{\mathbf{w},\mathbf{b}}$ is a continuous piecewise linear function (linear spline) on any compact subset Ω of $\mathbb{R}^d$.

Our first main result verifies the universality of deep CNNs, asserting that any function $f \in C(\Omega)$, the space of continuous functions on Ω with norm $\|f\|_{C(\Omega)} = \sup_{x\in\Omega} |f(x)|$, can be approximated by $\mathcal{H}_J^{\mathbf{w},\mathbf{b}}$ to an arbitrary accuracy when the depth J is large enough.

Theorem 1. *Let $2 \le s \le d$. For any compact subset Ω of $\mathbb{R}^d$ and any $f \in C(\Omega)$, there exist sequences $\mathbf{w}$ of filter masks, $\mathbf{b}$ of bias vectors and $f_J^{\mathbf{w},\mathbf{b}} \in \mathcal{H}_J^{\mathbf{w},\mathbf{b}}$ such that*

$$\lim_{J\to\infty} \|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} = 0.$$

Our second main result presents rates of approximation by deep CNNs for functions in the Sobolev space $H^r(\Omega)$ with an integer index $r > 2 + d/2$. Such a function f is the restriction to Ω of a function F from the Sobolev space $H^r(\mathbb{R}^d)$ on $\mathbb{R}^d$ meaning that F and all its partial derivatives up to order r are square integrable on $\mathbb{R}^d$.

Theorem 2. *Let $2 \le s \le d$ and $\Omega \subseteq [-1,1]^d$. If $J \ge 2d/(s-1)$ and $f = F|_\Omega$ with $F \in H^r(\mathbb{R}^d)$ and an integer index $r > 2 + d/2$, then there exist $\mathbf{w}$, $\mathbf{b}$ and $f_J^{\mathbf{w},\mathbf{b}} \in \mathcal{H}_J^{\mathbf{w},\mathbf{b}}$ such that*

$$\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} \le c\,\|F\|\,\sqrt{\log J}\,(1/J)^{\frac{1}{2}+\frac{1}{d}}, \tag{1.6}$$

where c is an absolute constant and $\|F\|$ denotes the Sobolev norm of $F \in H^r(\mathbb{R}^d)$.

According to Theorem 2, if we take $s = \lceil 1 + d^\tau/2 \rceil$ and $J = \lceil 4d^{1-\tau} \rceil L$ with $0 \le \tau \le 1$ and $L \in \mathbb{N}$, where $\lceil u \rceil$ denotes the smallest integer not smaller than u, then we have

$$\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} \le c\,\|F\| \sqrt{\frac{(1-\tau)\log d + \log L + \log 5}{4d^{1-\tau}L}},$$

while the widths of the deep CNN are bounded by $12Ld$ and the total number of free parameters by

$$(5s+2)J + 2d - 2s - 1 \le (73L+2)d.$$

We can even take $L = 1$ and $\tau = 1/2$ to get a bound for the relative error

$$\frac{\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)}}{\|F\|} \le \frac{c}{2} d^{-\frac{1}{4}} \sqrt{\log(5\sqrt{d})}$$

achieved by a deep CNN of depth $\lceil 4\sqrt{d} \rceil$ and at most $75d$ free parameters, which decreases as the dimension d increases. This interesting observation is new for deep CNNs, and does not exist in the literature of fully connected neural networks. It explains the strong approximation ability of deep CNNs.

A key contribution in our theory of deep CNNs is that an arbitrary pre-assigned sequence $W = (W_k)_{-\infty}^{\infty}$ supported in $\{0, \ldots, \mathcal{M}\}$ can be factorized into convolutions of a mask sequence $\{w^{(j)}\}_{j=1}^J$. It is proved by the same argument as in [30] for the case with the special restriction $W_0 \ne 0$. Convolutions are closely related to translation-invariance in speeches and images [3,6,27], and also in some learning algorithms [25,8].

Theorem 3. *Let $s \geq 2$ and $W = (W_k)_{-\infty}^{\infty}$ be a sequence supported in $\{0, \ldots, \mathcal{M}\}$ with $\mathcal{M} \geq 0$. Then there exists a finite sequence of filter masks $\{w^{(j)}\}_{j=1}^{J}$ supported in $\{0, \ldots, s\}$ with $J < \frac{\mathcal{M}}{s-1} + 1$ such that the convolutional factorization $W = w^{(J)} * \ldots * w^{(2)} * w^{(1)}$ holds true.*

2. Discussion

The classical shallow neural networks associated with an activation function $\sigma : \mathbb{R} \to \mathbb{R}$ produce functions of the form

$$f_N(x) = \sum_{k=1}^{N} c_k \sigma(\alpha_k \cdot x - b_k) \tag{2.1}$$

with $\alpha_k \in \mathbb{R}^d, b_k, c_k \in \mathbb{R}$. A mathematical theory for approximation of functions by shallow neural networks (2.1) was well developed three decades ago [5,12,1,20,17,23] and was extended to fully connected multi-layer neural networks shortly afterwards [12,20,4].

The first type of results obtained in the late 1980s are about universality, asserting that any continuous function f on any compact subset Ω of $\mathbb{R}^d$ can be approximated by some f_N to an arbitrary accuracy when the number of hidden neurons N is large enough. Such results were given in [5,12,1] when σ is a sigmoidal function, meaning that σ is a continuous strictly increasing function satisfying $\lim_{u\to-\infty} \sigma(u) = 0$ and $\lim_{u\to\infty} \sigma(u) = 1$. A more general result with a locally bounded and piecewise continuous activation function σ asserts [17,23] that universality holds if and only if σ is not a polynomial.

The second type of results obtained in the early 1990s are about rates of approximation. When σ is a C^∞ sigmoidal function and $f = F|_{[-1,1]^d}$ for some $F \in L^2(\mathbb{R}^d)$ with the Fourier transform $\hat{F}$ satisfying $|w|\hat{F}(w) \in L^1(\mathbb{R}^d)$, rates of type $\|f_N - f\|_{L^2_\mu([-1,1]^d)} = O(1/\sqrt{N})$ were given in [1] where μ is an arbitrary probability measure μ. Analysis was conducted in [20] for shallow neural networks (2.1) with more general continuous activation functions σ satisfying a special condition with some $b \in \mathbb{R}$ that $\sigma^{(k)}(b) \neq 0$ for any nonnegative integer k and a further assumption with some integer $\ell \neq 1$ that $\lim_{u\to-\infty} \sigma(u)/|u|^\ell = 0$ and $\lim_{u\to\infty} \sigma(u)/u^\ell = 1$. The rates there are of type $\|f_N - f\|_{C([-1,1]^d)} = O(N^{-r/d})$ for $f \in C^r([-1,1]^d)$. Note that the ReLU activation function considered in this paper does not satisfy the condition with $\sigma^{(k)}(b) \neq 0$ or the special assumption with $\ell \neq 1$. To achieve the approximation accuracy $\|f_N - f\|_{C([-1,1]^d)} \leq \epsilon$, when $r = \lceil \frac{d+1}{2} + 2 \rceil$ with $d/r \approx 2$, the number of hidden neurons $N \geq (c_{f,d,\ell}/\epsilon)^{d/r}$ and the total number of free parameters is at least $(c_{f,d,\ell}/\epsilon)^{d/r}\, d$, where the constant $c_{f,d,\ell}$ depends on the dimension d and might be very large.

To compare with our result, we take the filter length $s = \lceil 1 + d/2 \rceil$ and depth $J = 4L$ with $L \in \mathbb{N}$. We know from Theorem 2 that the same approximation accuracy $\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} \leq \epsilon$ with $0 < \epsilon \leq c\,\|F\|$ can be achieved by the deep CNN of depth $J = 4\lceil \frac{1}{\epsilon^2} \log \frac{1}{\epsilon^2} \rceil$ having at most $\lceil \frac{75}{\epsilon^2} \log \frac{1}{\epsilon^2} \rceil d$ free parameters, which does not depend on the dimension d. Though a logarithmic term is involved, this dimension independence gives evidence for the power of deep CNNs.

A multi-layer neural network is a sequence of function vectors $h^{(j)}(x)$ satisfying an iterative relation

$$h^{(j)}(x) = \sigma\left(T^{(j)} h^{(j-1)}(x) - b^{(j)}\right), \qquad j = 1, 2, \ldots, J. \tag{2.2}$$

Here $T^{(j)}$ is a full connection matrix without special structures. So a deep CNN is a special multi-layer neural network with sparse convolutional matrices. This sparsity gives difficulty in developing a mathematical theory for deep CNNs, since the techniques in the literature of fully connected shallow or multi-layer neural networks do not apply. Our novelty to overcome the difficulty is to factorize an arbitrary finitely supported sequence into convolutions of filter masks $\{w^{(j)}\}_{j=1}^{J}$ supported in $\{0, 1, \ldots, s\}$. Our method can be applied to distributed learning algorithms [18,10].

Recently there have been quite a few papers [28,7,29,24,2,22] on approximation and representation of functions by deep neural networks and benefit of depth, but all these results are for fully connected networks without pre-specified structures, not for deep CNNs. In particular, it was shown in [2,22] that the rate of approximation of some function classes by multi-layer fully connected neural networks (2.2) may be achieved by networks with sparse connection matrices $T^{(j)}$, but the locations of the sparse connections are unknown. This sparsity of unknown pattern is totally different from that of deep CNNs, the latter enables computing methods like stochastic gradient descent to learn values of the free parameters efficiently.

When the activation function is ReLU, explicit rates of approximation by fully connected neural networks were obtained recently in [13] for shallow nets, in [24] for nets with 3 hidden layers, and in [29,2,22] for nets with more layers. In particular, it was shown in Theorem 1 of [29] that for $f \in C^r([0,1]^d)$, the approximation accuracy $\epsilon \in (0,1)$ can be achieved by a ReLU deep net with at most $c(\log(1/\epsilon)+1)$ layers and at most $c\epsilon^{-d/r}(\log(1/\epsilon)+1)$ weights and computation units with a constant $c = c(d,r)$. To compare this result with ours when d is large, we need to derive an explicit lower bound for the number of parameters in the above net from the analysis in [29] which is based on Taylor polynomials of f and trapezoid functions defined by σ. With an integer parameter $N \in \mathbb{N}$, for $m \in \{0,1,\ldots,N\}^d$, the Taylor polynomial at m/N is $\sum_{\|\alpha\|_1<r} \frac{D^\alpha f(m/N)}{\alpha!}(x-m/N)^\alpha$, where $\alpha! = \Pi_{i=1}^d \alpha_i!$ and $(x-m/N)^\alpha = \Pi_{i=1}^d (x_i - m_i/N)^{\alpha_i}$ for $\alpha = (\alpha_i)_{i=1}^d \in \mathbb{Z}_+^d$. The trapezoid functions take the form $\phi_m(x) = \Pi_{i=1}^d \varphi(3Nx_i - m_i)$, where φ is the univariate trapezoid function $\varphi(u) = \sigma(u+2) - \sigma(u+1) - \sigma(u-1) + \sigma(u-2)$ supported on $[-2,2]$, equal to 1 on $[-1,1]$, and linear on $[-2,-1] \cup [1,2]$. The functions $\{\phi_m(x) : m \in \{0,1,\ldots,N\}^d\}$ form a partition of unity with $\sum_{m\in\{0,1,\ldots,N\}^d} \phi_m(x) \equiv 1$ on $[0,1]^d$, so f can be approximated by

$$\sum_{m\in\{0,1,\ldots,N\}^d} \sum_{\|\alpha\|_1<r} \frac{D^\alpha f(m/N)}{\alpha!}\phi_m(x)(x-m/N)^\alpha \tag{2.3}$$

with an accuracy $\frac{2^d d^r}{r!}N^{-r}$. By a key construction in [29], each basis function $\phi_m(x)(x-m/N)^\alpha$ in (2.3) can be approximated with an accuracy $(d+r)\delta$ by a ReLU net of depth at least $c_1(d+\|\alpha\|_1)\log(1/\delta)$ with a constant $c_1 = c_1(d,r)$ bounded from below by an absolute constant $C_0 > 0$. Thus, to achieve an accuracy $\epsilon \in (0,1)$ for approximating f by a ReLU deep net, one takes $N = \lceil \left(\frac{2^{d+1}d^r}{\epsilon r!}\right)^{1/r} \rceil$ and $\delta = \frac{\epsilon}{2^{d+1}d^r(d+r)}$ as in [29] and know that the depth of the net is at least $\frac{C_0 d}{2}(\log(1/\epsilon) + (d+1)\log 2 + r\log d + \log(d+r))$, and the total number of parameters for the net is more than the number of coefficients $\frac{D^\alpha f(m/N)}{\alpha!}$ which is

$$(N+1)^d \begin{pmatrix} d+r-1 \\ d \end{pmatrix} > \left(\frac{2^{d+1}d^r}{\epsilon r!}\right)^{d/r} > \epsilon^{-d/r}\left(\frac{2^{\frac{d+1}{r}}d}{r}\right)^d.$$

If we take $r = \lceil \frac{d+1}{2} + 2 \rceil$ as in our previous discussion, we see that for $d \geq 6$, the deep net constructed in Theorem 1 of [29] for achieving an accuracy $\epsilon \in (0,1)$ for approximating $f \in C^r([0,1]^d)$ has at least $2^d\epsilon^{-d/r}$ free parameters and at least $\frac{C_0 d}{4}(\log(1/\epsilon)+d)$ layers. When d is large and ϵ is fixed, this number of free parameters with an exponential factor 2^d is much larger than that of the deep CNN derived from Theorem 2.

Deep CNNs are often combined with pooling, a small number of fully connected layers, and some other techniques for improving the practical performance of deep learning. Our purpose to analyze purely convolutional networks is to demonstrate that convolution makes full use of shift-invariance properties of speeches and images for extracting data features efficiently. Also, for processing an image, convolutions based on the 2-D lattice $\mathbb{Z}^2$ are implemented by taking inner products of $(s+1)\times(s+1)$ filter matrices with shifted patches of the image. Though we do not consider such deep learning algorithms in this paper, some of our ideas can be used to establish mathematical theories for more general deep neural networks involving convolutions.

3. Proof of main results

For approximation in $C(\Omega)$ we can only consider those Sobolev spaces which can be embedded into the space of continuous functions, that is, those spaces with the regularity index $r > \frac{d}{2}$. To establish rates of approximation we require $r > \frac{d}{2}+2$ in Theorem 2. In this case, the set $H^r(\Omega)$ is dense in $C(\Omega)$, so Theorem 1 follows from Theorem 2 by scaling.

Proof of Theorem 2. Let $J \geq \frac{2d}{s-1}$ and m be the integer part of $\frac{(s-1)J}{d}-1 \geq 1$. In our assumption, $f = F|_\Omega$ for some function $F \in H^r(\mathbb{R}^d)$ with the Fourier transform $\widehat{F}(\omega)$ giving the norm $\|F\| = \left\| \left(1+|\omega|^2\right)^{r/2} \widehat{F}(\omega) \right\|_{L^2}$. By the Schwarz inequality and the condition $r > \frac{d}{2}+2$, $v_{F,2} := \int_{\mathbb{R}^d} \|\omega\|_1^2 \left|\widehat{F}(\omega)\right| d\omega \leq c_{d,r} \|F\|$ where $c_{d,r}$ is the finite constant $\left\| \|\omega\|_1^2 \left(1+|\omega|^2\right)^{-r/2} \right\|_{L^2}$. Then we apply a recent result from [13] on ridge approximation to $F|_{[-1,1]^d}$ and know that there exists a linear combination of ramp ridge functions of the form

$$F_m(x) = \beta_0 + \alpha_0 \cdot x + \frac{v}{m} \sum_{k=1}^{m} \beta_k \left(\alpha_k \cdot x - t_k\right)_+$$

with $\beta_k \in [-1,1], \|\alpha_k\|_1 = 1, t_k \in [0,1], \beta_0 = F(0), \alpha_0 = \nabla F(0)$ and $|v| \leq 2v_{F,2}$ such that

$$\|F - F_m\|_{C([-1,1]^d)} \leq c_0 v_{F,2} \max\left\{\sqrt{\log m}, \sqrt{d}\right\} m^{-\frac{1}{2}-\frac{1}{d}}$$

for some universal constant $c_0 > 0$.

Now we turn to the key step of constructing the filter mask sequence $\mathbf{w}$. Define a sequence W supported in $\{0,\ldots,(m+1)d-1\}$ by stacking the vectors $\alpha_0, \alpha_1, \ldots, \alpha_m$ (with components reversed) by

$$\left[W_{(m+1)d-1} \ \ldots \ W_1 \ W_0\right] = \left[\alpha_m^T \ \ldots \ \alpha_1^T \ \alpha_0^T\right].$$

We apply Theorem 3 to the sequence W with support in $\{0,1,\ldots,(m+1)d\}$ and find a sequence of filter masks $\mathbf{w} = \{w^{(j)}\}_{j=1}^{\hat{J}}$ supported in $\{0,1,\ldots,s\}$ with $\hat{J} < \frac{(m+1)d}{s-1}+1$ such that $W = w^{(\hat{J})} * w^{(\hat{J}-1)} * \ldots * w^{(2)} * w^{(1)}$. The choice of m implies $\frac{(m+1)d}{s-1} \leq J$. So $\hat{J} \leq J$ and by taking $w^{(\hat{J}+1)} = \ldots = w^{(J)}$ to be the delta sequence, we have $W = w^{(J)} * w^{(J-1)} * \ldots * w^{(2)} * w^{(1)}$. This tells us [30] that

$$T^{(J)} \ldots T^{(1)} = T^W$$

where T^W is the $d_J \times d$ matrix given by $[W_{\ell-k}]_{\ell=1,\ldots,d_J, k=1,\ldots,d}$. Observe from the definition of the sequence W that for $k = 0,1,\ldots,m$, the $(k+1)d$-th row of T^W is exactly the transpose of α_k. Also, since $Js \geq (m+1)d$, we have $W_{Js} = 0$ and the last row of T^W is a zero row.

Then we construct $\mathbf{b}$. Denote $\|w\|_1 = \sum_{k=-\infty}^{\infty} |w_k|$, $B^{(0)} = \max_{x\in\Omega} \max_{k=1,\ldots,d} |x_k|$ and $B^{(j)} = \|w^{(j)}\|_1 \ldots \|w^{(1)}\|_1 B^{(0)}$ for $j \geq 1$. Then we have

$$\left\| \left(T^{(j)} \ldots T^{(1)} x\right)_k \right\|_{C(\Omega)} \leq B^{(j)}, \qquad \forall k = 1,\ldots,d_j.$$

Take $b^{(1)} = -B^{(1)} \mathbf{1}_{d_1} := -B^{(1)}(1,\ldots,1)^T$, and

$$b^{(j)} = B^{(j-1)} T^{(j)} \mathbf{1}_{d_{j-1}} - B^{(j)} \mathbf{1}_{d_j}, \qquad j = 1,\ldots,J-1.$$

Then for $j = 1,\ldots,J-1$, we have

$$h^{(j)}(x) = T^{(j)} \dots T^{(1)} x + B^{(j)} \mathbf{1}_{d_j}$$

and $b_\ell^{(j)} = B^{(j-1)} \sum_{k=0}^{s} w_k^{(j)} - B^{(j)} = b_{s+1}^{(j)}$ for $\ell = s+1, \dots, d_j - s$. Hence the bias vectors are of the required form (1.4).

Finally, we take the bias vector $b^{(J)}$ by setting $b_\ell^{(J)}$ to be

$$\begin{cases} B^{(J-1)}(T^{(J)}\mathbf{1}_{d_{J-1}})_\ell - B^{(J)}, & \text{if } \ell = d, d + Js, \\ B^{(J-1)}(T^{(J)}\mathbf{1}_{d_{J-1}})_\ell + t_k, & \text{if } \ell = (k+1)d,\ 1 \le k \le m, \\ B^{(J-1)}(T^{(J)}\mathbf{1}_{d_{J-1}})_\ell + B^{(J)}, & \text{otherwise.} \end{cases}$$

Substituting this bias vector and the expression for $h^{(J-1)}(x)$ into the iterative relation of the deep CNN, we see from the identity $T^{(J)} \dots T^{(1)} = T^W$ and the definition of the sequence W that the ℓ-th component $h_\ell^{(J)}(x)$ of $h^{(J)}(x)$ equals

$$\begin{cases} \alpha_0 \cdot x + B^{(J)}, & \text{if } \ell = d, \\ B^{(J)}, & \text{if } \ell = d + Js, \\ (\alpha_k \cdot x - t_k)_+, & \text{if } \ell = (k+1)d,\ 1 \le k \le m, \\ 0, & \text{otherwise.} \end{cases}$$

Thus, we can take $f_J^{\mathbf{w},\mathbf{b}} = F_m|_\Omega \in \operatorname{span}\{h_k^{(J)}(x)\}_{k=1}^{d_J} = \mathcal{H}_J^{\mathbf{w},\mathbf{b}}$ and know that the error $\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} \le \|F - F_m\|_{C([-1,1]^d)}$ can be bounded as

$$\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} \le c_0 v_{F,2} \max\left\{\sqrt{\log m}, \sqrt{d}\right\} m^{-\frac{1}{2}-\frac{1}{d}}.$$

But $\frac{1}{2}(s-1)J \le md < (s-1)J$ and $2r - d - 4 \ge 1$. By a polar coordinate transformation, $c_{d,r} d^{1+\frac{1}{d}} \le \sqrt{\frac{d^6 \pi^{d/2}}{\Gamma(\frac{d}{2}+1)}} \left(1 + \frac{1}{\sqrt{2r-d-4}}\right)$ which can be bounded by an absolute constant $c' := \max_{\ell \in \mathbb{N}} 2\sqrt{\ell^6 \pi^{\ell/2}/\Gamma(\frac{\ell}{2}+1)}$. Therefore,

$$\|f - f_J^{\mathbf{w},\mathbf{b}}\|_{C(\Omega)} \le 2c_0 c' \|F\| \sqrt{\log J} J^{-\frac{1}{2}-\frac{1}{d}}.$$

This proves Theorem 2 by taking $c = 2c_0 c'$.

Convolutional factorizations have been considered in our recent work [30] for sequences W supported in $\{0, 1, \dots, S\}$ with $S \ge d$ under the special restrictions $W_0 > 0$ and $W_S \ne 0$. Theorem 3 gives a more general result by improving the bound for J in [30] and removing the special restrictions on W_0 and W_S.

Proof of Theorem 3. We apply a useful concept from the literature of wavelets [6], the symbol $\widetilde{w}$ of a sequence w finitely supported in the set of nonnegative integers, defined as a polynomial on $\mathbb{C}$ by $\widetilde{w}(z) = \sum_{k=0}^{\infty} w_k z^k$. The symbol of the convoluted sequence $a * b$ is given by $\widetilde{a*b}(z) = \widetilde{a}(z)\widetilde{b}(z)$. Notice that the symbol $\widetilde{W}$ of the sequence W supported in $\{0, \dots, \mathcal{M}\}$ is a polynomial of degree M with real coefficients for some $0 \le M \le \mathcal{M}$. So we know that complex roots $z_k = x_k + iy_k$ of $\widetilde{W}$ with $x_k \ne 0$ appear in pairs and by $(z - z_k)(z - \overline{z_k}) = z^2 - 2x_k z + (x_k^2 + y_k^2)$, the polynomial $\widetilde{W}(z)$ can be completely factorized as

$$\widetilde{W}(z) = W_M \Pi_{k=1}^{K} \left\{z^2 - 2x_k z + (x_k^2 + y_k^2)\right\} \Pi_{k=2K+1}^{M} (z - x_k),$$

where $2K$ is the number of complex roots with multiplicity, and $M - 2K$ is the number of real roots with multiplicity. By taking groups of up to $s/2$ quadratic factors (or $(s-1)/2$ quadratic factors with a linear factor) and s linear factors in the above factorization, we get $\widetilde{W}(z) = \widetilde{w^{(J)}}(z) \dots \widetilde{w^{(2)}}(z)\widetilde{w^{(1)}}(z)$, a

factorization of $\widetilde{W}$ into polynomials of degree up to s, which yields a desired convolutional factorization $W = w^{(J)} * w^{(J-1)} * \ldots * w^{(2)} * w^{(1)}$ and proves Theorem 3.

Acknowledgments

The author would like to thank Gilbert Strang, Steve Smale, and the anonymous referees for their detailed suggestions and encouragement. The work described in this paper is supported partially by the Research Grants Council of Hong Kong [Project No. CityU 11303915] and by National Nature Science Foundation of China [Grant No. 11461161006].

References

[1] A. Barron, Universal approximation bounds for superpositions of a sigmoidal function, IEEE Trans. Inform. Theory 39 (1993) 930–945.
[2] H. Bölcskei, P. Grohs, G. Kutyniok, P. Petersen, Optimal approximation with sparsely connected deep neural networks, SIAM J. Math. Data Sci. 1 (2019) 8–45.
[3] J. Bruna, S. Mallat, Invariant scattering convolution networks, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2013) 1872–1886.
[4] C. Chui, X. Li, H. Mhaskar, Limitations of the approximation capabilities of neural networks with one hidden layer, Adv. Comput. Math. 5 (1996) 233–243.
[5] G. Cybenko, Approximations by superpositions of sigmoidal functions, Math. Control Signals Systems 2 (1989) 303–314.
[6] I. Daubechies, Ten Lectures on Wavelets, SIAM, 1992.
[7] R. Eldan, O. Shamir, The power of depth for feedforward neural networks, in: COLT, 2016, pp. 907–940.
[8] J. Fan, T. Hu, Q. Wu, D. Zhou, Consistency analysis of an empirical minimum error entropy algorithm, Appl. Comput. Harmon. Anal. 41 (2016) 164–189.
[9] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.
[10] Z. Guo, S. Lin, D.X. Zhou, Learning theory of distributed spectral algorithms, Inverse Probl. 33 (2017) 074009, 29 pp.
[11] G. Hinton, S. Osindero, Y. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (2006) 1527–1554.
[12] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Netw. 2 (1989) 359–366.
[13] J. Klusowski, A. Barron, Approximation by combinations of ReLU and squared ReLU ridge functions with ℓ^1 and ℓ^0 controls, IEEE Trans. Inform. Theory 64 (2018) 7649–7656.
[14] A. Krizhevsky, I. Sutskever, G. Hinton, Imagenet classification with deep convolutional neural networks, in: NIPS, 2012, pp. 1097–1105.
[15] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436.
[16] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (1998) 2278–2324.
[17] M. Leshno, Y. Lin, A. Pinkus, S. Schocken, Multilayer feedforward networks with a non-polynomial activation function can approximate any function, Neural Netw. 6 (1993) 861–867.
[18] S. Lin, X. Guo, D.X. Zhou, Distributed learning with regularized least squares, J. Mach. Learn. Res. 18 (2017) 1–31.
[19] S. Mallat, Understanding deep convolutional networks, Philos. Trans. R. Soc. A 374 (2016) 20150203.
[20] H. Mhaskar, Approximation properties of a multilayered feedforward artificial neural network, Adv. Comput. Math. 1 (1993) 61–80.
[21] H. Mhaskar, T. Poggio, Deep vs. shallow networks: an approximation theory perspective, Anal. Appl. 14 (2016) 829–848.
[22] P. Petersen, V. Voigtlaender, Optimal approximation of piecewise smooth functions using deep ReLU neural networks, Neural Netw. 108 (2018) 296–330.
[23] A. Pinkus, Approximation theory of the MLP model in neural networks, Acta Numer. 8 (1999) 143–195.
[24] U. Shaham, A. Cloninger, R. Coifman, Provable approximation properties for deep neural networks, Appl. Comput. Harmon. Anal. 44 (2018) 537–557.
[25] S. Smale, D.X. Zhou, Shannon sampling and function reconstruction from point values, Bull. Amer. Math. Soc. 41 (2004) 279–305.
[26] I. Steinwart, P. Thomann, N. Schmid, Learning with hierarchical Gaussian kernels, arXiv:1612.00824v1, 2016.
[27] G. Strang, Linear Algebra and Learning from Data, Wellesley-Cambridge Press, 2019.
[28] M. Telgarsky, Benefits of depth in neural networks, in: 29th Annual Conference on Learning Theory, in: PMLR, vol. 49, 2016, pp. 1517–1539.
[29] D. Yarotsky, Error bounds for approximations with deep ReLU networks, Neural Netw. 94 (2017) 103–114.
[30] D.X. Zhou, Deep distributed convolutional neural networks: universality, Anal. Appl. 16 (2018) 895–919.