Journal of Multivariate Analysis 182 (2021) 104695

Contents lists available at ScienceDirect

Journal of Multivariate Analysis

journal homepage: www.elsevier.com/locate/jmva

Analysis of the rate of convergence of fully connected deep neural network regression estimates with smooth activation function

Sophie Langer

Fachbereich Mathematik, Technische Universität Darmstadt, Schlossgartenstr. 7, 64289 Darmstadt, Germany

ARTICLE INFO

Article history:
Received 12 October 2020
Received in revised form 29 October 2020
Accepted 29 October 2020
Available online 10 November 2020

AMS 2010 subject classifications:
primary 62G08
secondary 41A25
82C32

Keywords:
Curse of dimensionality
Deep learning
Neural networks
Nonparametric regression
Rate of convergence

ABSTRACT

This article contributes to the current statistical theory of deep neural networks (DNNs). It was shown that DNNs are able to circumvent the so-called curse of dimensionality in case that suitable restrictions on the structure of the regression function hold. In most of those results the tuning parameter is the *sparsity* of the network, which describes the number of non-zero weights in the network. This constraint seemed to be the key factor for the good rate of convergence results. Recently, the assumption was disproved. In particular, it was shown that simple fully connected DNNs can achieve the same rate of convergence. Those fully connected DNNs are based on the unbounded ReLU activation function. In this article we extend the results to smooth activation functions, i.e., to the sigmoid activation function. It is shown that estimators based on fully connected DNNs with sigmoid activation function also achieve the minimax rates of convergence (up to $\ln n$-factors). In our result the number of hidden layers is fixed, the number of neurons per layer tends to infinity for sample size tending to infinity and a bound for the weights in the network is given.

1. Introduction

Deep neural networks (DNNs) have been shown great success in various tasks like pattern recognition and nonparametric regression (see, e.g., the monographs [1,7,10,13,14,23]). Unfortunately, little is yet known about why this method is so successful in practical applications. In particular, there is still a gap between the practical use and the theoretical understanding, which have to be filled to provide a method which is efficient and reliable. This article is inspired to contribute to the current statistical theory of DNNs. The most convenient way to do this is to analyze DNNs in the context of nonparametric regression.

1.1. *Nonparametric regression*

In nonparametric regression a $\mathbb{R}^d \times \mathbb{R}$-valued random vector $(\mathbf{X}, Y)$ satisfying $\mathbf{E}\{Y^2\} < \infty$ is considered. Given a sample of size n of $(\mathbf{X}, Y)$, i.e., given a data set

$$\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\},$$

E-mail address: langer@mathematik.tu-darmstadt.de.

https://doi.org/10.1016/j.jmva.2020.104695

where $(\mathbf{X}, Y)$, $(\mathbf{X}_1, Y_1)$, …, $(\mathbf{X}_n, Y_n)$ are i.i.d., the aim is to construct an estimator

$$m_n(\cdot) = m_n(\cdot, \mathcal{D}_n) : \mathbb{R}^d \to \mathbb{R}$$

of the so-called regression function $m : \mathbb{R}^d \to \mathbb{R}$, $m(\mathbf{x}) = \mathbf{E}\{Y|\mathbf{X} = \mathbf{x}\}$ such that the so-called L_2 error

$$\int |m_n(\mathbf{x}) - m(\mathbf{x})|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x})$$

is "small" (cf., e.g., [10] for a systematic introduction to nonparametric regression and a motivation for the L_2 error).

1.2. Neural networks

In order to construct such regression estimators with DNNs, the first step is to define a suitable space of functions $f : \mathbb{R}^d \to \mathbb{R}$ by using neural networks. The starting point here is the choice of an activation function $\sigma : \mathbb{R} \to \mathbb{R}$. Traditionally, so-called squashing functions are chosen as activation function $\sigma : \mathbb{R} \to \mathbb{R}$, which are nondecreasing and satisfy $\lim_{x\to-\infty} \sigma(x) = 0$ and $\lim_{x\to\infty} \sigma(x) = 1$, e.g., the so-called sigmoid activation function

$$\sigma(x) = \frac{1}{1 + \exp(-x)}, \quad x \in \mathbb{R}. \tag{1}$$

Recently, also unbounded activation functions are used, e.g., the ReLU activation function $\sigma(x) = \max\{x, 0\}$.

The network architecture $(L, \mathbf{k})$ depends on a positive integer L called the number of hidden layers and a width vector $\mathbf{k} = (k_1, \ldots, k_L) \in \mathbb{N}^L$ that describes the number of neurons in the first, second, …, Lth hidden layer. A feedforward DNN with network architecture $(L, \mathbf{k})$ and sigmoid activation function σ is a real-valued function defined on $\mathbb{R}^d$ of the form

$$f(\mathbf{x}) = \sum_{i=1}^{k_L} c_{1,i}^{(L)} f_i^{(L)}(\mathbf{x}) + c_{1,0}^{(L)} \tag{2}$$

for some $c_{1,0}^{(L)}, \ldots, c_{1,k_L}^{(L)} \in \mathbb{R}$ and for $f_i^{(L)}$'s recursively defined by

$$f_i^{(s)}(\mathbf{x}) = \sigma\left(\sum_{j=1}^{k_{s-1}} c_{i,j}^{(s-1)} f_j^{(s-1)}(\mathbf{x}) + c_{i,0}^{(s-1)}\right) \tag{3}$$

for some $c_{i,0}^{(s-1)}, \ldots, c_{i,k_{s-1}}^{(s-1)} \in \mathbb{R}$, $s \in \{2, \ldots, L\}$, and

$$f_i^{(1)}(\mathbf{x}) = \sigma\left(\sum_{j=1}^{d} c_{i,j}^{(0)} x^{(j)} + c_{i,0}^{(0)}\right) \tag{4}$$

for some $c_{i,0}^{(0)}, \ldots, c_{i,d}^{(0)} \in \mathbb{R}$. The space of DNNs with L hidden layers, r neurons per layer and all coefficients bounded by α is defined by

$$\mathcal{F}(L, r, \alpha) = \{f \ : f \text{ is of the form (2) with } k_1 = k_2 = \cdots = k_L = r, \ |c_{i,j}^{(\ell)}| \le \alpha \text{ for all } i, j, \ell\}. \tag{5}$$

Since the networks of this function space are only defined by its width and depth (and by a bound for the weights in the network) we refer to this function space, as in [29] and [18] as fully connected DNNs.

1.3. Least squares estimator

A corresponding estimator can then be defined with the principle of least squares. In particular, we choose $L = L_n$ hidden layers, a number $r = r_n$ of neurons per hidden layer and bound $\alpha = \alpha_n$ for all coefficients in the network in dependence to the sample size. The fully connected DNN regression estimator is then defined as the minimizer of the so-called empirical L_2 risk over the function space $\mathcal{F}(L_n, r_n, \alpha_n)$, which results in

$$m_n(\cdot) = \arg \min_{f \in \mathcal{F}(L_n, r_n, \alpha_n)} \frac{1}{n} \sum_{i=1}^{n} |f(\mathbf{X}_i) - Y_i|^2.$$

For simplicity we assume here and in the sequel that the minimum above indeed exists. When this is not the case our theoretical results also hold for any estimator which minimizes the above empirical L_2 risk up to a small additional term.

1.4. Curse of dimensionality

In order to judge the quality of such estimators theoretically, usually the rate of convergence of the L_2 error is considered. It is well-known, that smoothness assumptions on the regression function are necessary in order to derive non-trivial results on the rate of convergence (see, e.g., Theorem 7.2 and Problem 7.2 in [7] and Section 3 in [8]). For that purpose, we introduce the following definition of (p, C)-smoothness.

Definition 1. Let $p = q + s$ for some $q \in \mathbb{N}_0$ and $0 < s \leq 1$. A function $m : \mathbb{R}^d \to \mathbb{R}$ is called (p, C)-smooth, if for every $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}_0^d$ with $\sum_{j=1}^d \alpha_j = q$ the partial derivative $\partial^q m/(\partial x_1^{\alpha_1} \ldots \partial x_d^{\alpha_d})$ exists and satisfies

$$\left| \frac{\partial^q m}{\partial x_1^{\alpha_1} \ldots \partial x_d^{\alpha_d}}(\mathbf{x}) - \frac{\partial^q m}{\partial x_1^{\alpha_1} \ldots \partial x_d^{\alpha_d}}(\mathbf{z}) \right| \leq C \|\mathbf{x} - \mathbf{z}\|^s$$

for all $\mathbf{x}, \mathbf{z} \in \mathbb{R}^d$, where $\| \cdot \|$ denotes the Euclidean norm.

For this function space the optimal minimax rate of convergence in nonparametric regression is given by

$$n^{-2p/(2p+d)}$$

(see, e.g., [26]). This rate suffers from a characteristic feature in case of high-dimensional functions: If d is relatively large compared to p, then this rate of convergence can be extremely slow. This phenomenon is well-known as the curse of dimensionality and the only way to circumvent it is by imposing additional assumptions on the regression function. [27,28] assumed some additive structure on the regression function and showed some optimal minimax rate of convergence independent of the input dimension d. Other classes like a so-called single index models, in which

$$m(\mathbf{x}) = g(a^\top \mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^d$$

is assumed to hold, where $g : \mathbb{R} \to \mathbb{R}$ is a univariate function and $a \in \mathbb{R}^d$ is a d-dimensional vector were considered in [11,12,20,30]. Related to this is the so-called projection pursuit, where the regression function is assumed to be a sum of functions of the above form, i.e.,

$$m(\mathbf{x}) = \sum_{k=1}^{K} g_k(a_k^\top \mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^d$$

for $K \in \mathbb{N}$, $g_k : \mathbb{R} \to \mathbb{R}$ and $a_k \in \mathbb{R}^d$ (see, e.g., [9]). If we assume that the univariate functions in these postulated structures are (p, C)-smooth, adequately chosen regression estimators can achieve the above univariate rates of convergence up to some logarithmic factor (cf., e.g., Chapter 22 in [10]). [15] studied the case of a regression function, which satisfies

$$m(\mathbf{x}) = g\left(\sum_{\ell_1=1}^{L_1} g_{\ell_1} \left(\sum_{\ell_2=1}^{L_2} g_{\ell_1,\ell_2} \left(\ldots \sum_{\ell_r=1}^{L_r} g_{\ell_1,\ldots,\ell_r}(\mathbf{x}^{\ell_1,\ldots,\ell_r}) \right) \right) \right),$$

where $g, g_{\ell_1}, \ldots, g_{\ell_1,\ldots,\ell_r} : \mathbb{R} \to \mathbb{R}$ are (p, C)-smooth univariate functions and $\mathbf{x}^{\ell_1,\ldots,\ell_r}$ are single components of $\mathbf{x} \in \mathbb{R}^d$ (not necessarily different for two different indices $(\ell_1, \ldots, \ell_r)$). With the use of a penalized least squares estimator, they proved that in this setting the rate $n^{-2p/(2p+1)}$ can be achieved.

1.5. Related results for DNNs

The rate of convergence of neural networks regression estimators has been analyzed by [3–6,16,17,22,25]. For the L_2 error of a single hidden layer neural network, [5] proves a dimensionless rate of $n^{-1/2}$ (up to some logarithmic factor), provided the Fourier transform has a finite first moment (which basically requires that the function becomes smoother with increasing dimension d of $\mathbf{X}$). [22] showed a rate of $n^{(-2p/(2p+d+5))+\varepsilon}$ for the L_2 error of suitably defined single hidden layer neural network estimator for (p, C)-smooth functions, but their study was restricted to the use of a certain cosine squasher as the activation function. The rate of convergence of neural network regression estimators based on two layer neural networks has been analyzed in [16]. Therein, interaction models were studied, where the regression function satisfies

$$m(\mathbf{x}) = \sum_{I \subseteq \{1,\ldots,d\}, |I|=d^*} m_I(\mathbf{x}_I), \qquad \mathbf{x} \in \mathbb{R}^d$$

for some $d^* \in \{1, \ldots, d\}$ and $m_I : \mathbb{R}^{d^*} \to \mathbb{R}$ $(I \subseteq \{1, \ldots, d\}, |I| \leq d^*)$, where

$$\mathbf{x}_{\{i_1,\ldots,i_{d^*}\}} = (x^{(i_1)}, \ldots, x^{(i_{d^*})}) \quad \text{for } 1 \leq i_1 < \cdots < i_{d^*} \leq d,$$

and in case that all m_I are (p, C)-smooth for some $p \leq 1$ it was shown that suitable neural network estimators achieve a rate of convergence of $n^{-2p/(2p+d^*)}$ (up to some logarithmic factor), which is again a convergence rate independent of d. In [17], this result was extended to so-called (p, C)-smooth generalized hierarchical interaction models of order d^*, which are defined as follows:

Definition 2. Let $d \in \mathbb{N}$, $d^* \in \{1, \dots, d\}$ and $m : \mathbb{R}^d \to \mathbb{R}$.
(a) We say that m satisfies a generalized hierarchical interaction model of order d^* and level 0, if there exist $a_1, \dots, a_{d^*} \in \mathbb{R}^d$ and $f : \mathbb{R}^{d^*} \to \mathbb{R}$ such that

$$m(\mathbf{x}) = f(a_1^\top \mathbf{x}, \dots, a_{d^*}^\top \mathbf{x}) \quad \text{for all } x \in \mathbb{R}^d.$$

(b) We say that m satisfies a generalized hierarchical interaction model of order d^* and level $\ell + 1$, if there exist $K \in \mathbb{N}$, $g_k : \mathbb{R}^{d^*} \to \mathbb{R}$ $(k \in \{1, \dots, K\})$ and $f_{1,k}, \dots, f_{d^*,k} : \mathbb{R}^d \to \mathbb{R}$ $(k \in \{1, \dots, K\})$ such that $f_{1,k}, \dots, f_{d^*,k}$ $(k \in \{1, \dots, K\})$ satisfy a generalized hierarchical interaction model of order d^* and level ℓ and

$$m(\mathbf{x}) = \sum_{k=1}^{K} g_k \left(f_{1,k}(\mathbf{x}), \dots, f_{d^*,k}(\mathbf{x})\right) \quad \text{for all } \mathbf{x} \in \mathbb{R}^d.$$

(c) We say that the generalized hierarchical interaction model defined above is (p, C)-smooth, if all functions occurring in its definition are (p, C)-smooth according to Definition 1.

It was shown that for such models suitably defined multilayer neural networks (in which the number of hidden layers depends on the level of the generalized interaction model) achieve the rate of convergence $n^{-2p/(2p+d^*)}$ (up to some logarithmic factor) in case $p \leq 1$. [6] showed that this result even holds for $p > 1$ provided the squashing function is suitably chosen. Similar rate of convergence results as in [6] have been shown in [25] for neural network regression estimates using the ReLU activation function. Here slightly more general function spaces, which fulfill some composition assumption, were studied. [18] generalized the function space to so-called hierarchical composition models, i.e., functions which fulfill the following definition.

Definition 3. Let $d \in \mathbb{N}$ and $m : \mathbb{R}^d \to \mathbb{R}$.
(a) We say that m satisfies a hierarchical composition model of level 0 with order and smoothness constraint $\mathcal{P}$, if there exists a $K \in \{1, \dots, d\}$ such that

$$m(\mathbf{x}) = x^{(K)} \quad \text{for all } \mathbf{x} \in \mathbb{R}^d.$$

(b) We say that m satisfies a hierarchical composition model of level $\ell + 1$ with order and smoothness constraint $\mathcal{P}$, if there exist $(p, K) \in \mathcal{P}$, $C > 0$, $g : \mathbb{R}^K \to \mathbb{R}$ and $f_1, \dots, f_K : \mathbb{R}^d \to \mathbb{R}$, such that g is (p, C)-smooth, $f_1, \dots, f_K$ satisfy a hierarchical composition model of level ℓ with order and smoothness constraint $\mathcal{P}$ and

$$m(\mathbf{x}) = g(f_1(\mathbf{x}), \dots, f_K(\mathbf{x})) \quad \text{for all } \mathbf{x} \in \mathbb{R}^d$$

1.6. Fully connected DNNs

[18] showed for simple fully connected DNN regression estimators with ReLU activation function a rate of convergence of $\max_{(p,K)\in\mathcal{P}} n^{-2p/(2p+K)}$. The networks regarded therein are only defined by its width and depth and contrary to [6] and [25] no further sparsity constraint is needed. Reversely, this means, that not the number of nonzero weights, but the number of overall weights of the network is restricted. We see two main advantages in restricting a network in this sense: First, the characterization of a network by its width and depth (and therefore by its overall number of weights) implies the ones in terms of the nonzero weights, while it is not true the other way around. An example is given in Figs. 1 and 2 for the network class $\mathcal{F}(2, 5, \alpha)$.

Here we see that both, sparsely connected and fully connected networks, are contained in the network class $\mathcal{F}(2, 5, \alpha)$, while a network with full connectivity (between neurons of consecutive layers) as in Fig. 1 is not contained in a network class where the number of nonzero weights is restricted by 20. Second, the easy topology of the networks enables us an easy and fast implementation of a corresponding estimator. For instance, as shown in Listing 1, we can easily implement a least squares DNN regression estimator with the help of Python's packages `tensorflow` and `keras`. Remark that this example already uses the sigmoid activation function which fits to the theoretical results of this article.

Listing 1: Python code for fitting of fully connected neural networks to data x_{learn} and y_{learn}

```
model = Sequential()
model.add(Dense(d, activation="sigmoid", input_shape=(d,)))
for i in np.arange(L):
    model.add(Dense(K, activation="sigmoid"))
model.add(Dense(1))
model.compile(optimizer="adam",
                    loss="mean_squared_error")
model.fit(x=x_learn,y=y_learn)
```

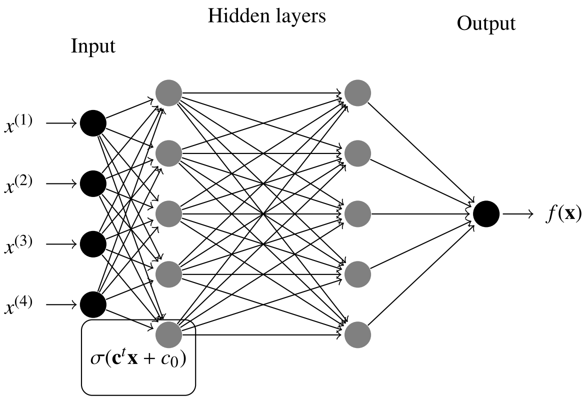

Fig. 1. A fully connected network of the class $\mathcal{F}(2, 5, \alpha)$ defined as in (5).

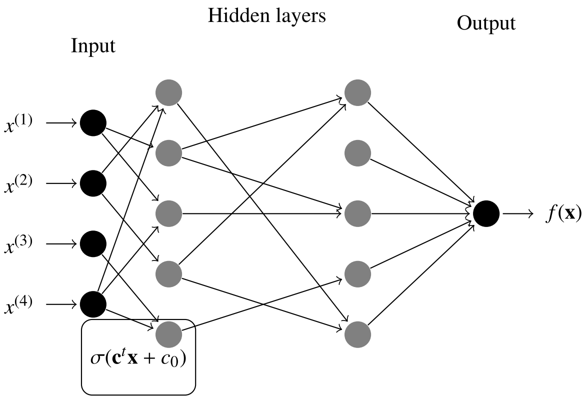

Fig. 2. A sparsely connected network of the class $\mathcal{F}(2, 5, \alpha)$ defined as in (5).

1.7. Main result in this article

[18] analyzes networks with ReLU activation function. We question ourselves if we can show the same rate of convergence for fully connected DNN regression estimators with smooth activation function. In this article we show that this is the case. In particular, we show that we derive a similar rate of convergence as in [6,18,25] for simple fully connected DNNs with sigmoid activation function. In these networks the number of hidden layer is fixed, the number of neurons per layer tends to infinity for sample size tending to infinity and a bound for the weights in the network is given. In the proofs the approximation results presented in [21] are essential.

1.8. Notation and outline

Throughout the paper, the following notation is used: The sets of natural numbers, natural numbers including 0 and real numbers are denoted by $\mathbb{N}$, $\mathbb{N}_0$ and $\mathbb{R}$, respectively. For $z \in \mathbb{R}$, we denote the smallest integer greater than or equal to z by $\lceil z \rceil$, and set $z_+ = \max\{z, 0\}$. Vectors are denoted by bold letters, e.g., $\mathbf{x} = (x^{(1)}, \dots, x^{(d)})^T$. We define $\mathbf{1} = (1, \dots, 1)^T$ and $\mathbf{0} = (0, \dots, 0)^T$. A d-dimensional multi-index is a d-dimensional vector $\mathbf{j} = (j^{(1)}, \dots, j^{(d)})^T \in \mathbb{N}_0^d$. As usual, we define

$$\|\mathbf{j}\|_1 = j^{(1)} + \cdots + j^{(d)}, \quad \mathbf{x}^{\mathbf{j}} = (x^{(1)})^{j^{(1)}} \cdots (x^{(d)})^{j^{(d)}}, \quad \mathbf{j}! = j^{(1)}! \cdots j^{(d)}!, \quad \partial^{\mathbf{j}} = \frac{\partial^{j^{(1)}}}{\partial (x^{(1)})^{j^{(1)}}} \cdots \frac{\partial^{j^{(d)}}}{\partial (x^{(d)})^{j^{(d)}}}.$$

Let $D \subseteq \mathbb{R}^d$ and let $f : \mathbb{R}^d \to \mathbb{R}$ be a real-valued function defined on $\mathbb{R}^d$. We write $\mathbf{x} = \arg\min_{\mathbf{z} \in D} f(\mathbf{z})$ if $\min_{\mathbf{z} \in \mathcal{D}} f(\mathbf{z})$ exists and if $\mathbf{x}$ satisfies $\mathbf{x} \in D$ and $f(\mathbf{x}) = \min_{\mathbf{z} \in \mathcal{D}} f(\mathbf{z})$. The Euclidean and the supremum norms of $\mathbf{x} \in \mathbb{R}^d$ are denoted by $\|\mathbf{x}\|$ and $\|\mathbf{x}\|_\infty$, respectively. For $f : \mathbb{R}^d \to \mathbb{R}$

$$\|f\|_\infty = \sup_{\mathbf{x} \in \mathbb{R}^d} |f(\mathbf{x})|$$

is its supremum norm, and the supremum norm of f on a set $A \subseteq \mathbb{R}^d$ is denoted by

$$\|f\|_{\infty,A} = \sup_{\mathbf{x}\in A} |f(\mathbf{x})|.$$

Furthermore we define $\|\cdot\|_{C^q(A)}$ of the smooth function space $C^q(A)$ by

$$\|f\|_{C^q(A)} := \max\left\{\|\partial^{\mathbf{j}} f\|_{\infty,A} : \|\mathbf{j}\|_1 \leq q, \mathbf{j} \in \mathbb{N}^d\right\}$$

for any $f \in C^q(A)$.

Let $A \subseteq \mathbb{R}^d$, let $\mathcal{F}$ be a set of functions $f : \mathbb{R}^d \to \mathbb{R}$ and let $\epsilon > 0$. A finite collection $f_1, \dots, f_N : \mathbb{R}^d \to \mathbb{R}$ is called an $\epsilon - \|\cdot\|_{\infty,A}-$ cover of $\mathcal{F}$ if for any $f \in \mathcal{F}$ there exists $i \in \{1, \dots, N\}$ such that

$$\|f - f_i\|_{\infty,A} = \sup_{\mathbf{x}\in A} |f(\mathbf{x}) - f_i(\mathbf{x})| < \epsilon.$$

The $\epsilon - \|\cdot\|_{\infty,A}-$ covering number of $\mathcal{F}$ is the size N of the smallest $\epsilon - \|\cdot\|_{\infty,A}-$ cover of $\mathcal{F}$ and is denoted by $\mathcal{N}(\epsilon, \mathcal{F}, \|\cdot\|_{\infty,A})$. We define the truncation operator T_β with level $\beta > 0$ as

$$T_\beta u = \begin{cases} u & \text{if} \quad |u| \leq \beta \\ \beta \operatorname{sign}(u) & \text{otherwise.} \end{cases}$$

The main result is presented in Section 2. Section 3 deals with a result concerning the approximation of a hierarchical composition model by neural networks. Section 4 contains the proof of the main result.

2. Main result

For $\ell = 1$ and some order and smoothness constraint $\mathcal{P} \subseteq (0, \infty) \times \mathbb{N}$ we define our space of hierarchical composition models by

$$\begin{aligned}\mathcal{H}(1, \mathcal{P}) = \{&h : \mathbb{R}^d \to \mathbb{R} : h(\mathbf{x}) = g(x^{(\pi(1))}, \dots, x^{(\pi(K_1^{(1)}))}),\\ &g : \mathbb{R}^{K_1^{(1)}} \to \mathbb{R} \text{ is } (p_1^{(1)}, C) \text{ –smooth for some } (p_1^{(1)}, K_1^{(1)}) \in \mathcal{P}, C > 0,\ \pi : \{1, \dots, K_1^{(1)}\} \to \{1, \dots, d\}\}.\end{aligned}$$

For $\ell > 1$, it recursively becomes

$$\begin{aligned}\mathcal{H}(\ell, \mathcal{P}) := \{&h : \mathbb{R}^d \to \mathbb{R} : h(\mathbf{x}) = g(f_1(\mathbf{x}), \dots, f_{K_1^{(\ell)}}(\mathbf{x})),\\ &g : \mathbb{R}^{K_1^{(\ell)}} \to \mathbb{R} \text{ is } (p_1^{(\ell)}, C)\text{–smooth for some } (p_1^{(\ell)}, K_1^{(\ell)}) \in \mathcal{P},\ C > 0,\ f_i \in \mathcal{H}(\ell - 1, \mathcal{P})\}.\end{aligned}$$

In practice, it is conceivable, that there exist input–output-relationships, which can be described by a regression function contained in $\mathcal{H}(\ell, \mathcal{P})$. Particularly, our assumption is motivated by applications in connection with complex technical systems, which are constructed in a modular form. Here each modular part can be again a complex system, which also explains the recursive construction in Definition 3. With regard to other function classes studied in the literature this function class generalizes previous results, as the function class of [6] (see Definition 2) forms some special case of $\mathcal{H}(\ell, \mathcal{P})$ in form of an alternation between summation and composition. Compared to the function class studied in [25], our definition forms a slight generalization, since we allow different smoothness and order constraints within the same level in the composition. We can now state the main result.

Theorem 1. *Let $(\boldsymbol{X}, Y), (\boldsymbol{X}_1, Y_1), \dots, (\boldsymbol{X}_n, Y_n)$ be independent and identically distributed random variables with values in $\mathbb{R}^d \times \mathbb{R}$ such that* $\operatorname{supp}(\boldsymbol{X})$ *is bounded and*

$$\mathrm{E}\left\{\exp(c_1 Y^2)\right\} < \infty$$

for some constant $c_1 > 0$. Let the corresponding regression function m be contained in the class $\mathcal{H}(\ell, \mathcal{P})$ for some $\ell \in \mathbb{N}$ and $\mathcal{P} \subseteq [1, \infty) \times \mathbb{N}$. Each function g in the definition of m can be of different smoothness $p_g = q_g + s_g$ ($q_g \in \mathbb{N}_0$ and $s_g \in (0, 1]$) and of different input dimension K_g, where $(p_g, K_g) \in \mathcal{P}$. Denote by K_{max} the maximal input dimension and by $p_{\max}$ the maximal smoothness of one of the functions g. Assume that for each g all partial derivatives of order less than or equal to q_g are bounded, i.e.,

$$\|g\|_{C^{q_g}(\mathbb{R}^d)} \leq c_2$$

for some constant $c_2 > 0$. Let each g be Lipschitz continuous with Lipschitz constant $C_{Lip} \geq 1$. Set
(i) $L_n = \ell\,(8 + \lceil\log_2(\max\{K_{\max}, p_{\max} + 1\})\rceil)$
(ii) $r_n = 2^{K_{\max}} \tilde{N}_1 29 \binom{K_{\max} + p_{\max}}{p_{\max}} K_{\max}^2 p_{\max} \max_{(p,K)\in\mathcal{P}} n^{K/(2(2p+K))}$
(iii) $\alpha_n = n^{c_3}$
with $c_3 > 0$ sufficiently large. Let $\sigma : \mathbb{R} \to [0, 1]$ be the sigmoid activation function $1/(1 + \exp(-x))$. Let $\tilde{m}_n$ be the least squares estimator defined by

$$\tilde{m}_n(\cdot) = \arg \min_{h\in\mathcal{F}(L_n, r_n, \alpha_n)} \frac{1}{n} \sum_{i=1}^{n} |Y_i - h(\boldsymbol{X}_i)|^2$$

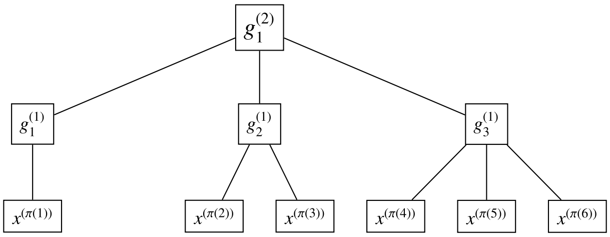

Fig. 3. Illustration of a hierarchical composition model of the class $\mathcal{H}(2,\mathcal{P})$ with the structure $h_1^{(2)}(\mathbf{x}) = g_1^{(2)}(h_1^{(1)}(\mathbf{x}), h_2^{(1)}(\mathbf{x}), h_3^{(1)}(\mathbf{x}))$, $h_1^{(1)}(\mathbf{x}) = g_1^{(1)}(x^{(\pi(1))})$, $h_2^{(1)}(\mathbf{x}) = g_2^{(1)}(x^{(\pi(2))}, x^{(\pi(3))})$ and $h_3^{(1)}(\mathbf{x}) = g_3^{(1)}(x^{(\pi(4))}, x^{(\pi(5))}, x^{(\pi(6))})$, defined as in (7) and (8).

and define $m_n = T_{c_4 \ln n}\tilde{m}_n$ for some $c_4 > 0$ sufficiently large. Then

$$\mathrm{E}\int |m_n(\boldsymbol{x}) - m(\boldsymbol{x})|^2 \mathrm{P}_{\boldsymbol{X}}(d\boldsymbol{x}) \leq c_5 (\ln n)^3 \max_{(p,K)\in\mathcal{P}} n^{-\frac{2p}{2p+K}}$$

holds for sufficiently large n.

Remark 1. Theorem 1 shows, that the L_2 errors of least squares neural network regression estimators based on a set of fully connected DNNs with a fixed number of layers (corresponding to a hierarchical composition model of given level ℓ and given smoothness and order constraint $\mathcal{P}$) achieves a rate of convergence $\max_{(p,K)\in\mathcal{P}} n^{-2p/(2p+K)}$ (up to some logarithmic factor), which does not depend on d and which does therefore circumvent the so-called *curse of dimensionality*.

Remark 2. Due to the fact that some parameters in the definition of the estimator in Theorem 1 are normally unknown in practice, they have to be chosen in a data-dependent way. Out of a set of different numbers of hidden layers and neurons per layer the best estimator is then chosen adaptively. Several possible methods and their effects can be found in [10].

3. Approximation of hierarchical composition models by DNNs

The aim of this section is to prove a result concerning the approximation of hierarchical composition models with smoothness and order constraint $\mathcal{P} \subseteq [1,\infty) \times \mathbb{N}$ by DNNs. In order to formulate this result, we observe in a first step, that one has to compute different hierarchical composition models of some level i ($i \in \{1,\ldots,\ell-1\}$) to compute a function $h_1^{(\ell)} \in \mathcal{H}(\ell,\mathcal{P})$. Let $\tilde{N}_i$ denote the number of hierarchical composition models of level i, needed to compute $h_1^{(\ell)}$. We denote in the following by

$$h_j^{(i)} : \mathbb{R}^d \to \mathbb{R} \tag{6}$$

the jth hierarchical composition model of some level i ($j \in \{1,\ldots,\tilde{N}_i\}$, $i \in \{1,\ldots,\ell\}$), that applies a $(p_j^{(i)}, C)$-smooth function $g_j^{(i)} : \mathbb{R}^{K_j^{(i)}} \to \mathbb{R}$ with $p_j^{(i)} = q_j^{(i)} + s_j^{(i)}$, $q_j^{(i)} \in \mathbb{N}_0$ and $s_j^{(i)} \in (0,1]$, where $(p_j^{(i)}, K_j^{(i)}) \in \mathcal{P}$. The computation of $h_1^{(\ell)}(x)$ can then be recursively described as follows:

$$h_j^{(i)}(\mathbf{x}) = g_j^{(i)}\left(h^{(i-1)}_{\sum_{t=1}^{j-1} K_t^{(i)}+1}(\mathbf{x}), \ldots, h^{(i-1)}_{\sum_{t=1}^{j} K_t^{(i)}}(\mathbf{x})\right) \tag{7}$$

for $j \in \{1,\ldots,\tilde{N}_i\}$ and $i \in \{2,\ldots,\ell\}$ and

$$h_j^{(1)}(\mathbf{x}) = g_j^{(1)}\left(x^{\left(\pi(\sum_{t=1}^{j-1} K_t^{(1)}+1)\right)}, \ldots, x^{\left(\pi(\sum_{t=1}^{j} K_t^{(1)})\right)}\right) \tag{8}$$

for some function $\pi : \{1,\ldots,\tilde{N}_1\} \to \{1,\ldots,d\}$. Furthermore for $i \in \{1,\ldots,\ell-1\}$ the recursion

$$\tilde{N}_l = 1, \ \tilde{N}_i = \sum_{j=1}^{\tilde{N}_{i+1}} K_j^{(i+1)} \tag{9}$$

holds.

The exemplary structure of a function $h_1^{(2)} \in \mathcal{H}(2,\mathcal{P})$ is illustrated in Fig. 3. Here one can get a perception of how the hierarchical composition models of different levels are stacked on top of each other. The approximation result of such a function $h_1^{(\ell)}$ by a DNN is summarized in the following theorem:

Theorem 2. *Let $m : \mathbb{R}^d \to \mathbb{R}$ be contained in the class $\mathcal{H}(\ell, \mathcal{P})$ for some $\ell \in \mathbb{N}$ and $\mathcal{P} \subseteq [1, \infty) \times \mathbb{N}$. Let $\tilde{N}_i$ be defined as in (9). Each m consists of different functions $h_j^{(i)}$ ($j \in \{1, \dots, \tilde{N}_i\}$, $i \in \{1, \dots, \ell\}$) defined as in (6), (7) and (8). Assume that the corresponding functions $g_j^{(i)}$ are Lipschitz continuous with Lipschitz constant $C_{Lip} \geq 1$ and satisfy*

$$\|g_j^{(i)}\|_{C^{q_j^{(i)}}([-a,a]^d)} \leq c_6$$

for some constant $c_6 > 0$. Denote by $K_{max} = \max_{i,j} K_j^{(i)}$ the maximal input dimension and by $p_{max} = \max_{i,j} p_j^{(i)}$ the maximal smoothness of the functions $g_j^{(i)}$. Then, for any $a \geq 1$, $M_{j,i} \in \mathbb{N}$ sufficiently large (each independent of the size of a, but $\min_{j,i} M_{j,i}^{2p_j^{(i)}} > c_7 \max\{\ell - 1(K_{\max} C_{Lip})^{\ell-2} a^{5p_{\max}+3}, 2^{K_{\max}}, 12K_{\max}\}$ must hold for some constants $c_7 > 0$ sufficiently large) and any
(i) $L = \ell\,(8 + \lceil \log_2(\max\{K_{\max}, p_{\max} + 1\})\rceil)$
(ii) $r = \max_{i \in \{1,\dots,\ell\}} \sum_{j=1}^{\tilde{N}_i} 29\binom{K_j^{(i)}+q_j^{(i)}}{q_j^{(i)}}(K_j^{(i)})^2(q_j^{(i)} + 1)M_{j,i}^{K_j^{(i)}}$
(iii) $\alpha = c_8 a^{22} e^{12 \times 2^{2(K_{\max}+1)+1} a K_{\max}} \max_{j,i} M_{j,i}^{32p_{\max}+4K_{\max}+18}$
a neural network $t \in \mathcal{F}(L, r, \alpha)$ exists such that

$$\|t - m\|_{\infty,[-a,a]^d} \leq c_9 a^{5p_{\max}+3} \max_{j,i} M_{j,i}^{-2p_j^{(i)}}.$$

In the proof of Theorem 2 we will need the following auxiliary results.

Lemma 1. *Let $\sigma : \mathbb{R} \to [0, 1]$ be the sigmoid activation function $\sigma(x) = 1/(1 + \exp(-x))$. Let $R \geq 1$ and $a > 0$. Then*

$$f_{id}(x) = 4R\left(\sigma\left(\frac{x}{R}\right) - 1\right) \in \mathcal{F}(1, 1, 4R)$$

satisfies for any $x \in [-a, a]$:

$$|f_{id}(x) - x| \leq 2\|\sigma''\|_\infty \frac{a^2}{R}.$$

Proof of Lemma 1. The result follows in a straightforward way from the proof of Theorem 2 in [24], cf., e.g., Lemma 1 in [19]. □

Lemma 2. *Let $1 \leq a < \infty$. Let $p = q + s$ for some $q \in \mathbb{N}_0$ and $s \in (0, 1]$, let $C \geq 1$. Let $m : \mathbb{R}^d \to \mathbb{R}$ be a (p, C)-smooth function, which satisfies*

$$\|m\|_{C^q([-2a,2a]^d)} \leq c_{10}.$$

for some constant $c_{10} > 0$. Let $\sigma : \mathbb{R} \to [0, 1]$ be the sigmoid activation function $1/(1 + \exp(-x))$. Then, for any $M \in \mathbb{N}$ sufficiently large (independent of the size of a, but $c_{11} \max\{a^{5p+3}, 2^d, 12d\} \leq M^{2p}$ must hold for some constant $c_{11} > 0$), a neural network $t \in \mathcal{F}(L, r, \alpha)$ with
(i) $L \geq 8 + \lceil \log_2(\max\{d, q + 1\})\rceil$
(ii) $r = 29\binom{d+q}{d}d^2(q + 1)M^d$
(iii) $\alpha = c_{12}\left(\max\left\{a, \|f\|_{C^q([-a,a]^d)}\right\}\right)^{11} e^{6 \times 2^{2(d+1)+1} ad} M^{16p+2d+9}$
exists such that

$$\|t - m\|_{\infty,[-a,a]^d} \leq c_{13} a^{5q+3} M^{-2p}$$

holds.

Proof of Lemma 2. For $L = 8 + \lceil \log_2(\max\{d, q + 1\})\rceil$ the proof follows directly from Theorem 1 in [21], where we use that

$$2^d\left(\max\left\{\left(\binom{d+q}{d} + d\right)M^d(2 + 2d) + d, 4(q + 1)\binom{d+q}{d}\right\} + M^d(2d + 2) + 12d\right) \leq 29\binom{d+q}{d}d^2(q + 1)M^d.$$

By successively applying f_{id} of Lemma 1 to the output of the network t, we can easily enlarge the number of hidden layers, such that the assertion also holds for $L > 8 + \lceil \log_2(\max\{d, q + 1\})\rceil$. Here we use that f_{id} satisfies

$$\left|f_{id}^s(x) - x\right| \leq \sum_{k=1}^{s}\left|f_{id}^k(x) - f_{id}^{k-1}(x)\right| = \sum_{k=1}^{s}\left|f_{id}(f_{id}^{k-1}(x)) - f_{id}^{k-1}(x)\right| \leq \frac{s}{M^{2p}}$$

for $s \in \mathbb{N}$ and $x \in \left[-2\max\left\{a, \|m\|_{\infty,[-a,a]^d}\right\}, 2\max\left\{a, \|m\|_{\infty,[-a,a]^d}\right\}\right]$, where we choose

$$R \geq (s - 1)8\|\sigma''\|_\infty \max\left\{a, \|m\|_{\infty,[-a,a]^d}\right\}^2 M^{2p}$$

in Lemma 1. Since t satisfies

$$\|t\|_{\infty,[-a,a]^d} \leq \|t-m\|_{\infty,[-a,a]^d} + \|m\|_{\infty,[-a,a]^b} \leq 2\max\left\{a, \|m\|_{C^q([-a,a]^d)}\right\},$$

where we use that $M^{2p} \geq c_{13}a^{5q+3}$, we can conclude that

$$\left|f^s(t(\mathbf{x})) - m(\mathbf{x})\right| \leq \left|f^s(t(\mathbf{x})) - t(\mathbf{x})\right| + |t(\mathbf{x}) - m(\mathbf{x})| \leq c_{13}a^{5q+3}M^{-2p}$$

holds for $\mathbf{x} \in [-a,a]^d$ and $s \in \mathbb{N}$. □

Proof of Theorem 2. The proof is divided into *two* steps.

Step 1: Network architecture: The computation of the function $m(\mathbf{x}) = h_1^{(\ell)}(\mathbf{x})$ can be recursively described as in (7) and (8). The basic idea of the proof is to define a composed network, which approximately computes the functions $h_1^{(1)}, \ldots, h_{\tilde{N}_1}^{(1)}, h_1^{(2)}, \ldots, h_{\tilde{N}_2}^{(2)}, \ldots, h_1^{(\ell)}$. For the approximation of $g_j^{(i)}$ we will use the networks

$$f_{net,g_j^{(i)}} \in \mathcal{F}(L_0, r_j^{(i)}, \alpha_0)$$

described in Lemma 2, where

$$L_0 = 8 + \lceil \log_2(\max\{K_{\max}, p_{\max}+1\}) \rceil,$$

$$r_j^{(i)} = 29\binom{K_j^{(i)} + q_j^{(i)}}{q_j^{(i)}}(K_j^{(i)})^2(q_j^{(i)}+1)M_{j,i}^{K_j^{(i)}}$$

and

$$\alpha_0 = c_8 a^{12} e^{6\times 2^{2(K_{\max}+1)+1} aK_{\max}} \max_{j,i} M_{j,i}^{10p_{\max}+2K_{\max}+10}$$

To compute the values of $h_1^{(1)}, \ldots, h_{\tilde{N}_1}^{(1)}$ we use the networks

$$\hat{h}_1^{(1)}(\mathbf{x}) = f_{net,g_1^{(1)}}\left(x^{(\pi(1))}, \ldots, x^{(\pi(K_1^{(1)}))}\right)$$

$$\vdots$$

$$\hat{h}_{\tilde{N}_1}^{(1)}(\mathbf{x}) = f_{net,g_{\tilde{N}_1}^{(1)}}\left(x^{(\pi(\sum_{t=1}^{\tilde{N}_1-1} K_t^{(1)}+1))}, \ldots, x^{(\pi(\sum_{t=1}^{\tilde{N}_1} K_t^{(1)}))}\right).$$

To compute the values of $h_1^{(i)}, \ldots, h_{\tilde{N}_i}^{(i)}$ ($i \in \{2, \ldots, \ell\}$) we use the networks

$$\hat{h}_j^{(i)}(\mathbf{x}) = f_{net,g_j^{(i)}}\left(\hat{h}^{(i-1)}_{\sum_{t=1}^{j-1} K_t^{(i)}+1}(\mathbf{x}), \ldots, \hat{h}^{(i-1)}_{\sum_{t=1}^{j} K_t^{(i)}}(\mathbf{x})\right)$$

for $j \in \{1, \ldots, \tilde{N}_i\}$. Finally we set

$$t(\mathbf{x}) = \hat{h}_1^{(\ell)}(\mathbf{x}).$$

Fig. 4 illustrates the computation of the network $t(\mathbf{x})$. It is easy to see that $t(\mathbf{x})$ forms a composed network, where the networks $\hat{h}_1^{(i)}, \ldots, \hat{h}_{\tilde{N}_i}^{(i)}$ are computed in parallel (i.e., in the same layers) for $i \in \{1, \ldots, \ell\}$, respectively. Since each $\hat{h}_j^{(i)}$ ($j \in \{1, \ldots, \tilde{N}_i\}$) needs L_0 layers, $r_j^{(i)}$ neurons per layer and has α_0 as bound for its weights, this network is contained in the class

$$\mathcal{F}\left(\ell L_0, \max_{i\in\{1,\ldots,\ell\}} \sum_{j=1}^{\tilde{N}_i} r_j^{(i)}, \alpha_0^2\right) \subseteq \mathcal{F}(L, r, \alpha).$$

Step 2: Approximation error: We define

$$g_{\max} := \max\left\{\max_{\substack{i\in\{1,\ldots,\ell\},\\ j\in\{1,\ldots,\tilde{N}_i\}}} \|g_j^{(i)}\|_\infty, 1\right\}.$$

Since each $g_j^{(i)}$ satisfies the assumption of Lemma 2, we can conclude that for $\mathbf{x} \in [-2\max\{g_{\max}, a\}, 2\max\{g_{\max}, a\}]^{K_j^{(i)}}$

$$\left|f_{net,g_j^{(i)}}(\mathbf{x}) - g_j^{(i)}(\mathbf{x})\right| \leq c_{14}a^{5p_{\max}+3}\max_{j,i} M_{j,i}^{-2p_j^{(i)}}, \tag{10}$$

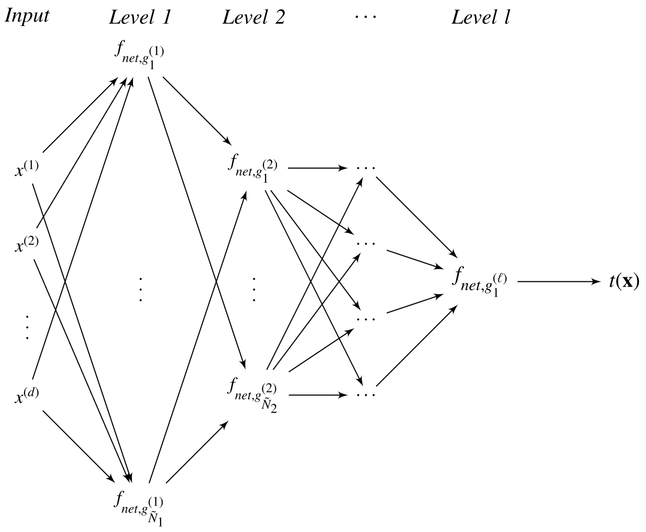

Fig. 4. Illustration of the DNN $t(\mathbf{x})$, which shows how the networks $f_{net,g_1^{(i)}}, \dots, f_{net,g_{\tilde{N}_i}^{(i)}}$ are computed in parallel for $i \in \{1, \dots, \ell\}$, respectively.

where

$$c_{14} = c_{13}(2g_{max})^{5p_{\max}+3}.$$

We show by induction that

$$\left|\hat{h}_j^{(i)}(\mathbf{x}) - h_j^{(i)}(\mathbf{x})\right| \le c_{14} i (K_{\max} C_{Lip})^{i-1} a^{5p_{\max}+3} \max_{j,i} M_{j,i}^{-2p_j^{(i)}}. \tag{11}$$

By (10) we can conclude that for $j \in \{1, \dots, \tilde{N}_1\}$

$$\left|\hat{h}_j^{(1)}(\mathbf{x}) - h_j^{(1)}(\mathbf{x})\right| \le c_{14} 1 (K_{\max} C_{Lip})^{1-1} a^{5p_{\max}+3} \max_{j,i} M_{j,i}^{-2p_j^{(i)}}.$$

Thus we have shown that (11) holds for $i = 1$. Assume now that (11) holds for some $i - 1$ and every $j \in \{1, \dots, \tilde{N}_{i-1}\}$. Then

$$\left|\hat{h}_j^{(i-1)}(\mathbf{x})\right| \le \left|\hat{h}_j^{(i-1)}(\mathbf{x}) - h_j^{(i-1)}(\mathbf{x})\right| + g_{\max} \le 2g_{\max}$$

follows directly by the induction hypothesis, where we use that $\min_{j,i} M_{j,i}^{2p_j^{(i)}} \ge c_{14}(i-1)(K_{\max}C_{Lip})^{i-1}a^{5p_{\max}+3}$. Using (10) and the Lipschitz continuity of $g_j^{(i)}$ we can conclude that

$$\begin{aligned}
\left|\hat{h}_j^{(i)}(\mathbf{x}) - h_j^{(i)}(\mathbf{x})\right| &= \left|f_{net,g_j^{(i)}}\left(\hat{h}^{(i-1)}_{\sum_{t=1}^{j-1} K_t^{(i)}+1}(\mathbf{x}), \dots, \hat{h}^{(i-1)}_{\sum_{t=1}^{j} K_t^{(i)}}(\mathbf{x})\right) - g_j^{(i)}\left(\hat{h}^{(i-1)}_{\sum_{t=1}^{j-1} K_t^{(i)}+1}(\mathbf{x}), \dots, \hat{h}^{(i-1)}_{\sum_{t=1}^{j} K_t^{(i)}}(\mathbf{x})\right)\right| \\
&\quad + \left|g_j^{(i)}\left(\hat{h}^{(i-1)}_{\sum_{t=1}^{j-1} K_t^{(i)}+1}(\mathbf{x}), \dots, \hat{h}^{(i-1)}_{\sum_{t=1}^{j} K_t^{(i)}}(\mathbf{x})\right) - g_j^{(i)}\left(h^{(i-1)}_{\sum_{t=1}^{j-1} K_t^{(i)}+1}(\mathbf{x}), \dots, h^{(i-1)}_{\sum_{t=1}^{j} K_t^{(i)}}(\mathbf{x})\right)\right| \\
&\le c_{14} a^{5p_{\max}+3} \max_{j,i} M_{j,i}^{-2p_j^{(i)}} + K_j^{(i)} C_{Lip} c_{14} (i-1)(K_{max}C_{Lip})^{i-2} a^{5p_{\max}+3} \max_{j,i} M_{j,i}^{-2p_j^{(i)}} \\
&\le c_{15} i (K_{max}C_{Lip})^{i-1} a^{5p_{\max}+3} \max_{j,i} M_{j,i}^{-2p_j^{(i)}}.
\end{aligned}$$

Thus we have shown that there exists a network t satisfying

$$\|t - m\|_{\infty,[-a,a]^d} \le c_9 a^{5p_{\max}+3} \max_{j,i} M_{j,i}^{-2p_j^{(i)}}.$$

This proves the assertion of the theorem. □

4. Proof of the main result

4.1. An auxiliary result from the empirical process theory

In the proof of Theorem 1 we use the following bound on the expected L_2 error of the least squares estimators.

Lemma 3. *Assume that the distribution of $(\boldsymbol{X}, Y)$ satisfies $\mathrm{E}\{\exp(c_{16}Y^2)\} < \infty$ for some constant $c_{16} > 0$ and that the regression function m is bounded in absolute value. Let $\tilde{m}_n$ be the least squares estimator*

$$\tilde{m}_n(\cdot) = \arg\min_{f \in \mathcal{F}_n} \frac{1}{n} \sum_{i=1}^{n} |Y_i - f(\boldsymbol{X}_i)|^2$$

based on some function space $\mathcal{F}_n$ and set $m_n = T_{c_{17}\ln n}\tilde{m}_n$ for some constant $c_{17} > 0$. Then m_n satisfies

$$\mathrm{E}\int |m_n(\boldsymbol{x}) - m(\boldsymbol{x})|^2 \mathrm{Pr}_{\boldsymbol{X}}(d\boldsymbol{x}) \le \frac{c_{18}(\ln n)^2 \left(\ln\left(\mathcal{N}\left(\frac{1}{nc_{17}\ln n}, \mathcal{F}_n, \|\cdot\|_{\infty,\mathrm{supp}(X)}\right)\right) + 1\right)}{n} + 2\inf_{f\in\mathcal{F}_n}\int |f(\boldsymbol{x}) - m(\boldsymbol{x})|^2 \mathrm{Pr}_{\boldsymbol{X}}(d\boldsymbol{x})$$

for $n > 1$ and some constant $c_{18} > 0$, which does not depend on n or the parameters of the estimator.

Proof. This proof follows in a straightforward way from the proof of Theorem 1 in [2]. A complete proof can be found in the supplement of [6]. □

4.2. A bound on the covering number

If the function class $\mathcal{F}_n$ in Lemma 3 forms a class of fully connected DNNs $\mathcal{F}(L, r, \alpha)$ with α and L bounded, the following result will help to bound the covering number:

Lemma 4. *Let $\epsilon \ge 1/n^{c_{19}}$ and let $\mathcal{F}(L, r, \alpha)$ defined as in (5) with $\sigma : \mathbb{R} \to [0, 1]$ Lipschitz continuous with Lipschitz constant $C_{Lip} > 0$, $1 \le \max\{a, \alpha\} \le n^{c_{20}}$ and $L \le c_{21}$ for large n and certain constants $c_{19}, c_{20}, c_{21} > 0$. Then*

$$\left(\ln \mathcal{N}(\epsilon, \mathcal{F}(L, r, \alpha), \|\cdot\|_{\infty,[-a,a]^d})\right) \le c_{22}(1 + \ln n + \ln r)(r + 1)^2$$

holds for sufficiently large n and a constant $c_{22} > 0$ independent of n.

Proof. Let

$$f(\mathbf{x}) = \sum_{i=1}^{r} c_{1,i}^{(L)} f_i^{(L)}(\mathbf{x}) + c_{1,0}^{(L)}, \quad \bar{f}(\mathbf{x}) = \sum_{i=1}^{r} \bar{c}_{1,i}^{(L)} \bar{f}_i^{(L)}(\mathbf{x}) + \bar{c}_{1,0}^{(L)},$$

for some $c_{1,0}^{(L)}, \bar{c}_{1,0}^{(L)}, \ldots, c_{1,r}^{(L)}, \bar{c}_{1,r}^{(L)} \in \mathbb{R}$ and for $f_i^{(L)}$'s, $\bar{f}_i^{(L)}$'s recursively defined by

$$f_i^{(s)}(\mathbf{x}) = \sigma\left(\sum_{j=1}^{r} c_{i,j}^{(s-1)} f_j^{(s-1)}(\mathbf{x}) + c_{i,0}^{(s-1)}\right), \quad \bar{f}_i^{(s)}(\mathbf{x}) = \sigma\left(\sum_{j=1}^{r} \bar{c}_{i,j}^{(s-1)} \bar{f}_j^{(s-1)}(\mathbf{x}) + \bar{c}_{i,0}^{(s-1)}\right)$$

for some $c_{i,0}^{(s-1)}, \bar{c}_{i,0}^{(s-1)}, \ldots, c_{i,r}^{(s-1)}, \bar{c}_{i,r}^{(s-1)} \in \mathbb{R}$, $s \in \{2, \ldots, L\}$, and

$$f_i^{(1)}(\mathbf{x}) = \sigma\left(\sum_{j=1}^{d} c_{i,j}^{(0)} x^{(j)} + c_{i,0}^{(0)}\right), \quad \bar{f}_i^{(1)}(\mathbf{x}) = \sigma\left(\sum_{j=1}^{d} \bar{c}_{i,j}^{(0)} x^{(j)} + \bar{c}_{i,0}^{(0)}\right)$$

for some $c_{i,0}^{(0)}, \bar{c}_{i,0}^{(0)}, \ldots, \bar{c}_{i,d}^{(0)} \in \mathbb{R}$. Let $C_{Lip} \ge 1$ be an upper bound on the Lipschitz constant of σ. Then

$$\begin{aligned}
|f(\mathbf{x}) - \bar{f}(\mathbf{x})| &\le \sum_{i=1}^{r} |c_{1,i}^{(L)}||f_i^{(L)}(\mathbf{x}) - \bar{f}_i^{(L)}(\mathbf{x})| + |c_{1,0}^{(L)} - \bar{c}_{1,0}^{(L)}| + \sum_{i=1}^{r} |c_{1,i}^{(L)} - \bar{c}_{1,i}^{(L)}||\bar{f}_i^{(L)}(\mathbf{x})| \\
&\le r \max_{i\in\{1,\ldots,r\}} |c_{1,i}^{(L)}| \max_{i\in\{1,\ldots,r\}} |f_i^{(L)}(\mathbf{x}) - \bar{f}_i^{(L)}(\mathbf{x})| + |c_{1,0}^{(L)} - \bar{c}_{1,0}^{(L)}| + r \max_{i\in\{1,\ldots,r\}} |c_i^{(L)} - \bar{c}_i^{(L)}|, \\
|f_i^{(s)}(x) - \bar{f}_i^{(s)}(\mathbf{x})| &\le C_{Lip} \left| \sum_{j=1}^{r} c_{i,j}^{(s-1)} f_j^{(s-1)}(\mathbf{x}) + c_{i,0}^{(s-1)} - \left(\sum_{j=1}^{r} \bar{c}_{i,j}^{(s-1)} \bar{f}_j^{(s-1)}(\mathbf{x}) + \bar{c}_{i,0}^{(s-1)}\right)\right| \\
&\le C_{Lip} r \max_{j\in\{1,\ldots,r\}} |c_{i,j}^{(s-1)}| \max_{j\in\{1,\ldots,r\}} |f_j^{(s-1)}(\mathbf{x}) - \bar{f}_j^{(s-1)}(\mathbf{x})| \\
&\quad + C_{Lip} r \max_{j\in\{1,\ldots,r\}} |c_{i,j}^{(s-1)} - \bar{c}_{i,j}^{(s-1)}| + C_{Lip}|c_{i,0}^{(s-1)} - \bar{c}_{i,0}^{(s-1)}|
\end{aligned}$$

for $s \in \{2, \dots, L\}$ and

$$|f_i^{(1)}(\mathbf{x}) - \bar{f}_i^{(1)}(\mathbf{x})| \le C_{Lip}(d+1) \max_{j \in \{0,\dots,d\}} |c_{i,j}^{(0)} - \bar{c}_{i,j}^{(0)}| a.$$

In the sequel we will use the abbreviation

$$\max_{i,j,s} |c_{i,j}^{(s)}| = \max\{\max_i |c_{1,i}^{(L)}|, \max_{i,j,s} |c_{i,j}^{(s)}|\}.$$

Recursively we conclude

$$\begin{aligned} |f(\mathbf{x}) - \bar{f}(\mathbf{x})| \le\ & (r+1) \max_{i \in \{0,\dots,r\}} |c_{1,i}^{(L)} - \bar{c}_{1,i}^{(L)}| + r \max_{i,j,s} |c_{i,j}^{(s)}| C_{Lip}(r+1) \max_{i,j} |c_{i,j}^{(L-1)} - \bar{c}_{i,j}^{(L-1)}| \\ & + r(\max_{i,j,s} |c_{i,j}^{(s)}| C_{Lip}(r+1))^2 \max_{i,j} |c_{i,j}^{(L-2)} - \bar{c}_{i,j}^{(L-2)}| + \cdots + r(\max_{i,j,s} |c_{i,j}^{(s)}| C_{Lip}(r+1))^{L-1} \max_{i,j} |c_{i,j}^{(1)} - \bar{c}_{i,j}^{(1)}| \\ & + r(\max_{i,j,s} |c_{i,j}^{(s)}| C_{Lip}(r+1))^{L-1} C_{Lip}(d+1) a \max_{i,j} |c_{i,j}^{(0)} - \bar{c}_{i,j}^{(0)}|. \end{aligned}$$

Provided we have

$$\max_{i \in \{0,\dots,r\}} |c_{1,i}^{(L)} - \bar{c}_{1,i}^{(L)}| \le \frac{\epsilon}{(L+1)(r+1)},$$

$$\max_{i,j} |c_{i,j}^{(t)} - \bar{c}_{i,j}^{(t)}| \le \frac{\epsilon}{(L+1) r (\max_{i,j,s} |c_{i,j}^{(s)}| C_{Lip}(r+1))^{L-t}}$$

for $t \in \{1, \dots, L\}$ and

$$\max_{i,j} |c_{i,j}^{(0)} - \bar{c}_{i,j}^{(0)}| \le \frac{\epsilon}{(L+1) r (\max_{i,j,s} |c_{i,j}^{(s)}| C_{Lip}(r+1))^{L-1} C_{Lip}(d+1) a}$$

implies

$$|f(\mathbf{x}) - \bar{f}(\mathbf{x})| \le \underbrace{\frac{\epsilon}{L+1} + \frac{\epsilon}{L+1} + \cdots + \frac{\epsilon}{L+1}}_{L+1-\text{times}} = \epsilon.$$

Assume that $c_{i,j}^{(s)}$ are all contained in the interval $[-\alpha, \alpha]$, where $\alpha \ge 1$. By discretizing this interval on the various levels s for each of the at most $(r+1)^2$ weights used in this level accordingly, we see that we can construct a supremum norm cover of size

$$\begin{aligned} & \prod_{t=0}^{L-1} \left(\frac{2\alpha(L+1)(r+1)(\alpha C_{Lip}(r+1))^t}{\epsilon} \right)^{(r+1)^2} \frac{2\alpha(L+1)(r+1)(\alpha C_{Lip}(r+1))^{L-1} C_{Lip}(d+1)a}{\epsilon} \\ & \le \left(\frac{2\alpha(L+1)(r+1)(\alpha C_{Lip}(r+1))^{L-1}}{\epsilon} \right)^{L(r+1)^2} \frac{2(L+1)(\alpha C_{Lip}(r+1))^L (d+1)a}{\epsilon} \\ & \le c_{28} \left(\frac{(L+1)(\alpha C_{Lip}(r+1))^L (d+1)a}{\epsilon} \right)^{(L+1)(r+1)^2}. \quad \Box \end{aligned}$$

4.3. *Proof of Theorem 1*

Let $a_n = (\ln n)^{3/(2\times(5p_{\max}+3))}$. For n sufficiently large the relation $\text{supp}(\mathbf{X}) \subseteq [-a_n, a_n]^d$ holds, which implies $\mathcal{N}(\delta, \mathcal{G}, \|\cdot\|_{\infty,\text{supp}(\mathbf{X})}) \le \mathcal{N}(\delta, \mathcal{G}, \|\cdot\|_{\infty,[-a_n,a_n]^d})$ for an arbitrary function space $\mathcal{G}$ and $\delta > 0$. Application of Lemma 3 leads to

$$\begin{aligned} & \mathbf{E} \int |m_n(\mathbf{x}) - m(\mathbf{x})|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x}) \\ & \le \frac{c_{18} (\ln n)^2 \left(\ln \left(\mathcal{N} \left(\frac{1}{n c_4 \ln n}, \mathcal{F}(L_n, r_n, \alpha_n), \|\cdot\|_{\infty,\text{supp}(\mathbf{X})} \right) \right) + 1 \right)}{n} + 2 \inf_{f \in \mathcal{F}(L_n r_n, \alpha_n)} \int |f(\mathbf{x}) - m(\mathbf{x})|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x}). \end{aligned}$$

Set

$$(\bar{p}, \bar{K}) = \arg\max_{(p,K) \in \mathcal{P}} n^{-\frac{2p}{2p+K}}.$$

The fact that $1/(nc_4 \ln n) \geq 1/n^{c_{19}}$, $\max\{a_n, \alpha\} \leq n^{c_{20}}$ and $r_n \leq c_{21} n^{1/2(2\bar{p}/\bar{K}+1)}$ holds for $c_{19}, c_{20}, c_{21} > 0$, allows us to apply Lemma 4 to bound the first summand by

$$\frac{c_{18}(\ln n)^2 c_{22}\left(1 + \ln n + \left(\ln c_{23} n^{\frac{1}{2(2\bar{p}/\bar{K}+1)}}\right)\right) c_{11} n^{\frac{1}{2\bar{p}/\bar{K}+1}}}{n} \leq \frac{c_{24}(\ln n)^3 n^{\frac{1}{2\bar{p}/\bar{K}+1}}}{n} \leq c_{24}(\ln n)^3 n^{-\frac{2\bar{p}}{2\bar{p}+\bar{K}}} \tag{12}$$

for a sufficiently large n. Regarding the second summand we apply Theorem 2, where we choose $M_{j,i} = \left\lceil n^{1/2(2p_j^{(i)}+K_j^{(i)})} \right\rceil$.

Since

$$\begin{aligned}
&\max_{i \in \{1,\ldots,\ell\}} \sum_{j=1}^{\tilde{N}_i} 29 \binom{K_j^{(i)} + q_j^{(i)}}{q_j^{(i)}} (K_j^{(i)})^2 (q_j^{(i)} + 1) M_{j,i}^{K_j^{(i)}} \\
\leq & \tilde{N}_1 29 \binom{K_{\max} + p_{\max}}{p_{\max}} K_{\max}^2 p_{\max} \max_{j,i} M_{j,i}^{K_j^{(i)}} = \tilde{N}_1 29 \binom{K_{\max} + p_{\max}}{p_{\max}} K_{\max}^2 p_{\max} \max_{j,i} \left\lceil n^{1/(2(2p_j^{(i)}+K_j^{(i)}))} \right\rceil^{K_j^{(i)}} \\
\leq & \tilde{N}_1 29 \binom{K_{\max} + p_{\max}}{p_{\max}} K_{\max}^2 p_{\max} \max_{j,i} \left(n^{1/(2(2p_j^{(i)}+K_j^{(i)}))} + 1 \right)^{K_j^{(i)}} \leq \tilde{N}_1 29 \binom{K_{\max} + p_{\max}}{p_{\max}} K_{\max}^2 p_{\max} \\
& \times \max_{j,i} \left(2n^{1/(2(2p_j^{(i)}+K_j^{(i)}))} \right)^{K_j^{(i)}} \\
\leq & 2^{K_{\max}} \tilde{N}_1 29 \binom{K_{\max} + p_{\max}}{p_{\max}} K_{\max}^2 p_{\max} \max_{j,i} n^{K_j^{(i)}/(2(2p_j^{(i)}+K_j^{(i)}))} = r
\end{aligned}$$

and

$$\begin{aligned}
\alpha &= c_8 a_n^{24} e^{12 \times 2^{2(K_{\max}+1)+1} a_n K_{\max}} \max_{j,i} M_{j,i}^{20p_{\max}+4K_{\max}+20} \\
&= c_8 \left((\ln n)^{3/(2\times(5p_{\max}+3))}\right)^{24} e^{12\times 2^{2(K_{\max}+1)+1}(\ln n)^{3/(2\times(5p_{\max}+3))} K_{\max}} \max_{j,i} \left\lceil n^{1/2(2p_j^{(i)}+K_j^{(i)})} \right\rceil^{20p_{\max}+4K_{\max}+20} \leq n^{c_{25}}
\end{aligned}$$

for $c_{25} > 0$ sufficiently large, the resulting values of r and α are consistent with r_n and α_n in Theorem 1. Theorem 2 allows us to bound $\inf_{f \in \mathcal{F}(L_n r_n, \alpha_n)} \int |f(\mathbf{x}) - m(\mathbf{x})|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x})$ by

$$c_{26} \left(a_n^{5p_{\max}+3}\right)^2 \max_{j,i} M_{j,i}^{-4p_j^{(i)}} = c_{26}(\ln n)^3 \max_{j,i} n^{-\frac{4p_j^{(i)}}{2(2p_j^{(i)}+K_j^{(i)})}}.$$

This together with (12) and the fact that

$$n^{-\frac{2\bar{p}}{2\bar{p}+\bar{K}}} = \max_{j,i} n^{-\frac{2p_j^{(i)}}{2p_j^{(i)}+K_j^{(i)}}}$$

implies the assertion.

References

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

[2] A.M. Bagirov, C. Clausen, M. Kohler, Estimation of a regression function by maxima of minima of linear functions, IEEE Trans. Inform. Theory 55 (2) (2009) 833–845.

[3] A.R. Barron, Complexity regularization with application to artificial neural networks, Nonparametr. Funct. Estim. Related Top. (1991) 561–576.

[4] A. Barron, Universal approximation bounds for superpositions of a sigmoidal function, IEEE Trans. Inform. Theory 39 (3) (1993) 930–945.

[5] A.R. Barron, Approximation and estimation bounds for artificial neural networks, Mach. Learn. 14 (1) (1994) 115–133.

[6] B. Bauer, M. Kohler, On deep learning as a remedy for the curse of dimensionality in nonparametric regression, Ann. Statist. 47 (2019) 2261–2285.

[7] L. Devroye, L. Györfi, G. Lugosi, A Probabilistic Theory of Pattern Recognition, Springer, 1996.

[8] L.P. Devroye, T.J. Wagner, Distribution-free consistency results in nonparametric discrimination and regression function estimation, Ann. Statist. 8 (2) (1980) 231–239.

[9] J.H. Friedman, W. Stuetzle, Projection pursuit regression, J. Amer. Statist. Assoc. 76 (376) (1981) 817–823.

[10] L. Györfi, M. Kohler, A. Krzyżak, H. Walk, A Distribution-Free Theory of Nonparametric Regression, Springer Series in Statistics, Springer, 2002.

[11] W. Härdle, P. Hall, H. Ichimura, Optimal smoothing in single-index models, Ann. Statist. 21 (1) (1993) 157–178.

[12] W. Härdle, T.M. Stoker, Investigating smooth multiple regression by the method of average derivatives, J. Amer. Statist. Assoc. 84 (408) (1989) 986–995.

[13] S. Haykin, Neural Networks: A Comprehensive Foundation, second ed., Prentice Hall PTR, Upper Saddle River, NJ, USA, 1998.

[14] J. Hertz, R.G. Palmer, A.S. Krogh, Introduction to the Theory of Neural Computation, first ed., Perseus Publishing, 1991.

[15] J.L. Horowitz, E. Mammen, Rate-optimal estimation for a general class of nonparametric regression models with unknown link functions, Ann. Statist. 35 (6) (2007) 2589–2619.
[16] M. Kohler, A. Krzyżak, Adaptive regression estimation with multilayer feedforward neural networks, J. Nonparametr. Stat. 17 (8) (2005) 891–913.
[17] M. Kohler, A. Krzyżak, Nonparametric regression based on hierarchical interaction models, IEEE Trans. Inform. Theory 63 (3) (2017) 1620–1630.
[18] M. Kohler, S. Langer, On the rate of convergence of fully connected deep neural network regression estimates, 2020, ArXiv preprint arxiv:1908.11133.
[19] M. Kohler, S. Langer, A. Krzyżak, Estimation of a function of low local dimensionality by deep neural networks, 2020, Arxiv preprint arxiv:1908.11140.
[20] E. Kong, Y. Xia, Variable selection for the single–index model, Biometrika 94 (1) (2007) 217–229.
[21] S. Langer, Approximating smooth functions by deep neural networks with sigmoidal activation function and fixed depth, 2020, Arxiv preprint arxiv:2010.04596.
[22] D.F. McCaffrey, A.R. Gallant, Convergence rates for single hidden layer feedforward networks, Neural Netw. 7 (1) (1994) 147–158.
[23] B.D. Ripley, N.L. Hjort, Pattern Recognition and Neural Networks, first ed., Cambridge University Press, New York, NY, USA, 1995.
[24] F. Scarselli, A.C. Tsoi, Universal approximation using feedforward neural networks: A survey of some existing methods, and some new results, Neural Netw. 11 (1) (1998) 15–37.
[25] J. Schmidt-Hieber, Nonparametric regression using deep neural networks with relu activation function, Ann. Statist. 48 (4) (2020) 1875–1897.
[26] C.J. Stone, Optimal global rates of convergence for nonparametric regression, Ann. Statist. 10 (4) (1982) 1040–1053.
[27] C.J. Stone, Additive regression and other nonparametric models, Ann. Statist. 13 (2) (1985) 689–705.
[28] C.J. Stone, The use of polynomial splines and their tensor products in multivariate function estimation, Ann. Statist. 22 (1) (1994) 118–171.
[29] D. Yarotsky, A. Zhevnerchuk, The phase diagram of approximation rates for deep neural networks, 2019, Arxiv preprint arXiv:1906.09477.
[30] Y. Yu, D. Ruppert, Penalized spline estimation for partially linear single-index models, J. Amer. Statist. Assoc. 97 (460) (2002) 1042–1054.