On the rate of convergence of an over-parametrized deep neural network regression estimate with ReLU activation function learned by gradient descent [*]

Michael Kohler[1] and Adam Krzyżak[2,†]

[1] *Fachbereich Mathematatik, Technische Universität Darmstadt, Schlossgartenstr. 7, 64289 Darmstadt, Germany, email: kohler@mathematik.tu-darmstadt.de*

[2] *Department of Computer Science and Software Engineering, Concordia University, 1455 De Maisonneuve Blvd. West, Montreal, Quebec, Canada H3G 1M8, email: krzyzak@cs.concordia.ca*

July 19, 2023

Abstract
Estimation of a regression function from independent and identically distributed random variables is considered. The L_2 error with integration with respect to the design measure is used as an error criterion. Over-parametrized deep neural network estimates with ReLU activation function are defined where all the weights are learned by the gradient descent. It is shown that the expected L_2 error of the estimates converges to zero with rate

$$n^{-\frac{p}{2p+d}}$$

(up to some logarithmic factor) in case that the regression function is p-times continuously differentiable. In case that the regression function satisfies the assumption of a p times continuously differentiable interaction model, i.e., in case that it is equal to a finite sum of functions where each function in the sum is a p-times continuously differentiable function applied to only d^* of the d components of its input, we show that our estimate achieves the above rate of convergence with d replaced by d^*.

AMS classification: Primary 62G08; secondary 62G20.

Key words and phrases: neural networks, nonparametric regression, over-parametrization, rate of convergence, ReLU activation function.

1 Introduction

1.1 Deep Learning

Deep neural networks are among the most successful approaches in multivariate statistical estimation applications and have been applied extremely successfully in many

[*] Running title: *Over-parametrized deep neural networks*

[†] Corresponding author. Tel: +1-514-848-2424 ext. 3007, Fax:+1-514-848-2830

different areas, e.g., in image classification (cf., e.g., Krizhevsky, Sutskever and Hinton (2012)), language recognition (cf., e.g., Kim (2014)) machine translation (cf., e.g., Wu et al. (2016)) or mastering of games (cf., e.g., Silver et al. (2017)). Motivated by the practical success of these networks there has been in the last 6 years an increasing interest in studying the corresponding estimators theoretically. As pointed out in Kutyonik (2020), the theoretical analysis of deep neural networks can be separated into three parts: expressivity, optimization, and generalization. Here in expressivity it is studied which functions can be approximated well by deep neural networks. In optimization it is investigated how a deep neural network can be fitted to observed data. And in generalization it is analyzed how well deep neural networks adapted to one data set behave on new independent data sets of the same kind.

The purpose of this paper is to extend the theoretical knowledge about deep neural networks by studying simultaneously expressivity, optimization, and generalization for over-parametrized deep neural networks estimates with ReLU activation function.

1.2 Nonparametric regression

To do this, we study deep neural networks in the context of nonparametric regression. Here, (X,Y) is an $\mathbb{R}^d \times \mathbb{R}$–valued random vector with $\mathbf{E}Y^2 < \infty$, and $m(x) = \mathbf{E}\{Y|X = x\}$ is the corresponding regression function $m : \mathbb{R}^d \to \mathbb{R}$. Given a sample of (X,Y), i.e., a data set

$$\mathcal{D}_n = \{(X_1,Y_1),\dots,(X_n,Y_n)\}\,, \tag{1}$$

where (X,Y), (X_1,Y_1), …, (X_n,Y_n) are i.i.d., the goal is to construct an estimator

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

of the corresponding regression function $m(x) = \mathbf{E}\{Y|X = x\}$ such that the so–called L_2 error

$$\int |m_n(x) - m(x)|^2 \mathbf{P}_X(dx)$$

is “small” (cf., e.g., Györfi et al. (2002) for a systematic introduction to nonparametric regression and a motivation for the L_2 error).

We are interested to investigate for given estimates m_n how quickly the expected L_2 error

$$\mathbf{E}\int |m_n(x) - m(x)|^2 \mathbf{P}_X(dx) \tag{2}$$

converges to zero. It is well-known, that without regularity assumptions on the smoothness of m it is not possible to derive nontrivial asymptotic bounds on (2) (cf., Theorem 7.2 and Problem 7.2 in Devroye, Györfi and Lugosi (1996) and Section 3 in Devroye and Wagner (1980)). In order to formulate such regularity assumptions we will use in this paper the notion of (p,C)–smoothness, which we introduce next.

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 $\alpha = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}_0^d$ with $\sum_{j=1}^d \alpha_j = q$ the partial derivative $\frac{\partial^q m}{\partial x_1^{\alpha_1} \ldots \partial x_d^{\alpha_d}}$ exists and satisfies*

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

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

1.3 Main results

In this paper we analyze deep neural networks with ReLU activation function learned by gradient descent. Here we propose a special topology of the network where the output of the network is defined as a linear combination of a huge number of fully connected deep neural networks of constant widths and logarithmic depths. We introduce special initialization of the weights, where the output weights are zero and all inner weights are generated with various uniform distributions. Then we perform a suitable large number of gradient descent steps with the stepsize equal to one divided by the number of steps. We show that the expected L_2 error of the truncated version of the resulting estimate converges to zero with the rate of convergence

$$n^{-\frac{p}{2p+d}}$$

(up to some logarithmic factor) in case that the regression function is (p, C)-smooth for some $p, C > 0$, where p might be arbitrary large. In case of an interaction model of order d^* (i.e., in case that the regression function is a sum of functions where each function in the sum depends only on d^* of the d components of x) we show that the above rate of convergence holds with d replaced by d^*. It is well-known that the above rate of convergence is not optimal and that there exists simple estimates which achieve a better rate of convergence (cf., Stone (1982, 1994)), however we see our result as an important step towards a general convergence theory for deep neural network estimates learned by gradient descent.

1.4 Discussion of related results

Expressivity and generalization of deep neural networks are nowadays relatively well understood. There exist quite a few approximation results for neural networks (cf., e.g., Yarotsky (2018), Yarotsky and Zhevnerchute (2019), Lu et al. (2020), Langer (2021) and the literature cited therein), and generalization of deep neural networks can either be analyzed within the framework of the classical VC theory (using e.g. the result of Bartlett et al. (2019) to bound the VC dimension of classes of neural networks) or in case of over-parametrized deep neural networks (where the number of free parameters adjusted to the observed data set is much larger than the sample size) using bounds on the Rademacher complexity (cf., e.g., Liang, Rakhlin and Sridharan (2015), Golowich, Rakhlin and Shamir (2019), Lin and Zhang (2019), Wang and Ma (2022) and the literature cited therein). By combining these results expressivity and generalization has been controlled simultaneously in case of not over-parametrized deep neural networks. E.g., it has been shown in the context of nonparametric regression, that least squares estimates based on deep neural network achieve a dimension reduction in a high-dimensional setting in case that the regression function is a composition of functions where each functions depends only on a few input variables (cf., Kohler and Krzyżak (2017), Bauer and Kohler (2019), Schmidt-Hieber (2020) and Kohler and Langer (2021)). The mathematical reason behind these results is that due to the network structure composition of neural networks leads again to a neural network. This enables to generalize approximation results for deep neural networks for function classes to results for compositions of those functions, which together with proper bounds on the generalization properties of classes of neural networks enables to analyze least squares regression estimates based on deep neural networks. Adaptation of deep neural network to especially weak smoothness assumptions was shown in Imaizumi and Fukamizu (2018), Suzuki (2018) and Suzuki and Nitanda (2019).

Less well understood is the optimization of deep neural networks. If we consider optimization separately from expressivity and generalization, the main question is why gradient descent (and its variants) lead to estimates with small empirical risk. As was shown e.g. in Zou et al. (2018), Du et al. (2019), Allen-Zhu, Li and Song (2019) and Kawaguchi and Huang (2019) application of gradient descent to over-parameterized deep neural networks leads to neural networks that (globally) minimize the considered empirical risk. However, as was shown in Kohler and Krzyżak (2021), the corresponding estimates do not behave well on new independent data. So the main question is why gradient descent (and its variants like stochastic gradient descent) can be used to fit a neural network to observed data in such a way that the resulting estimate achieves good results on a new independent data. The challenge here is not only to analyze the optimization but to consider it simultaneously with expressivity and generalization.

In case of shallow neural networks (i.e., neural networks with only one hidden layer) this has been done successfully in Braun et al. (2023). Here it was possible to show that the classical dimension free rate of convergence of Barron (1994) for estimation of a regression function where its Fourier transform has a finite moment can also be achieved by shallow neural networks learned by the gradient descent. The main idea here is that the gradient descent selects a subset of the neural network where the random initialization of the inner weights have lead to values with good approximation properties, and that it adjusts the outer weights for these neurons properly. A similar idea was also applied in Gonon (2021). Kohler and Krzyżak (2022) applied this idea in the context of over-parametrized deep neural networks where a linear combination of a huge number of deep neural networks of fixed size are computed in parallel. Here the gradient descent selects again a subset of the neural networks computed in parallel and chooses a proper linear combination of the networks. By using metric entropy bounds (cf., e.g., Birman and Solomnjak (1967) and Li, Gu and Ding (2021)) it is possible to control the generalization of the over-parametrized neural networks, and as a result a rate of convergence of order close to $n^{-1/(1+d)}$ (or $n^{1/(1+d^*)}$ in case of interaction models, where it is assumed that the regression function is a sum of functions applied to only d^* of the d components of the predictor variable) can be shown for Hölder-smooth regression function with Hölder exponent $p \in [1/2, 1]$. In all those results adjusting the inner weights with gradient descent is not important. In fact, Gonon (2021) does not do this at all, while Braun et al. (2023) and Kohler and Krzyżak (2022) use that the relevant inner weights do not move too far away from their starting values during gradient descent. Similar ideas have also been applied in Andoni et al. (2014) and Daniely (2017). This whole approach is related to random feature networks (cf., e.g., Huang, Chen and Siew (2006) and Rahimi and Recht (2008a, 2008b, 2009)), where the inner weights are chosen randomly and only the outer weights are learned during gradient descent. Yehudai and Shamir (2022) present a lower bound which implies that either the number of neurons or the absolute value of the coefficients must grow exponential in the dimension in order to learn a single ReLU neuron with random feature networks. But since Braun et al. (2023) was able to prove a useful rate of convergence result for networks similar to random feature networks, the practical relevance of this lower bound is not clear.

A survey of various results on over-parametrized deep neural network estimates learned by gradient descent can be found in Bartlett, Montanari and Rakhlin (2021). These results usually analyze the estimates in some asymptotically equivalent models (like the mean field approach in Mei, Montanari, and Nguyen (2018), Chizat and Bach (2018) or Nguyen and Pham (2020) or the neural tangent approach in Hanin and Nica (2019)). In contrast, in this paper we analyze directly the error of the estimate in a standard regression model. Here the analysis of the gradient descent is related to Shamir and Zhang (2012), and we derive a new result for Rademacher complexities (cf., Lemma 5 below) in order to control the generalization error of the estimate. One trick in this context is to use a projection step in gradient descent to ensure that the sum of the absolute values of the output weights remains properly bounded. Otherwise our proof strategy is similar to the one introduced in Kohler and Krzyżak (2022).

1.5 Notation

The sets of natural numbers, real numbers and nonnegative real numbers are denoted by $\mathbb{N}$, $\mathbb{R}$ 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 we set $z_+ = \max\{z, 0\}$ and $z_- = \max\{-z, 0\}$. The Euclidean norm of $x \in \mathbb{R}^d$ is denoted by $\|x\|$. For a closed and convex set $A \subseteq \mathbb{R}^d$ we denote by $Proj_A x$ that element $Proj_A x \in A$ with

$$\|x - Proj_A x\| = \min_{z \in A} \|x - z\|.$$

For $f : \mathbb{R}^d \to \mathbb{R}$

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

is its supremum norm.

For $\mathbf{j} = (j^{(1)}, \dots, j^{(d)}) \in \mathbb{N}_0^d$ we write

$$\|\mathbf{j}\|_1 = j^{(1)} + \cdots + j^{(d)}$$

and for $f : \mathbb{R}^d \to \mathbb{R}$ we set

$$\partial^{\mathbf{j}} f = \frac{\partial^{\|\mathbf{j}\|_j} f}{(\partial x^{(1)})^{j^{(1)}} \dots (\partial x^{(d)})^{j^{(d)}}}.$$

Let $\mathcal{F}$ be a set of functions $f : \mathbb{R}^d \to \mathbb{R}$, let $x_1, \dots, x_n \in \mathbb{R}^d$, set $x_1^n = (x_1, \dots, x_n)$ and let $p \geq 1$. A finite collection $f_1, \dots, f_N : \mathbb{R}^d \to \mathbb{R}$ is called an L_p ε–packing in $\mathcal{F}$ on x_1^n if $f_1, \dots, f_N \in \mathcal{F}$ and

$$\min_{1 \leq i < j \leq N} \left(\frac{1}{n} \sum_{k=1}^{n} |f_i(x_k) - f_j(x_k)|^p \right)^{1/p} \geq \varepsilon$$

hold. The L_p ε–packing number of $\mathcal{F}$ on x_1^n is the size N of the largest L_p ε–packing of $\mathcal{F}$ on x_1^n and is denoted by $\mathcal{M}_p(\varepsilon, \mathcal{F}, x_1^n)$.

For $z \in \mathbb{R}$ and $\beta > 0$ we define $T_\beta z = \max\{-\beta, \min\{\beta, z\}\}$. If $f : \mathbb{R}^d \to \mathbb{R}$ is a function then we set $(T_\beta f)(x) = T_\beta\,(f(x))$.

1.6 Outline

The over-parametrized deep neural network estimates considered in this paper are introduced in Section 2. The main results are presented in Section 3. Section 4 contains the proofs.

2 Definition of the estimate

2.1 Topology of the deep networks

Throughout the paper we let $\sigma(x) = \max\{x, 0\}$ be the ReLU activation function and we define the topology of our neural networks as follows: We let $K_n, L_n, r \in \mathbb{N}$ and $\beta_n \in \mathbb{R}_+$ be parameters of our estimate and using these parameters we set

$$f_{\mathbf{w}}(x) = \sum_{j=1}^{K_n} w_{1,1,j}^{(L_n)} \cdot T_{\beta_n}(f_{\mathbf{w},j,1}^{(L_n)}(x)) \tag{3}$$

for some $w_{1,1,1}^{(L_n)}, \dots, w_{1,1,K_n}^{(L_n)} \in \mathbb{R}$, where $f_{j,1}^{(L_n)} = f_{\mathbf{w},j,1}^{(L_n)}$ are recursively defined by

$$f_{k,i}^{(l)}(x) = f_{\mathbf{w},k,i}^{(l)}(x) = \sigma\left(\sum_{j=1}^{r} w_{k,i,j}^{(l-1)} \cdot f_{k,j}^{(l-1)}(x) + w_{k,i,0}^{(l-1)} \right) \tag{4}$$

for some $w_{k,i,0}^{(l-1)}, \dots, w_{k,i,r}^{(l-1)} \in \mathbb{R}$ $(l = 2, \dots, L_n)$ and

$$f_{k,i}^{(1)}(x) = f_{\mathbf{w},k,i}^{(1)}(x) = \sigma\left(\sum_{j=1}^{d} w_{k,i,j}^{(0)} \cdot x^{(j)} + w_{k,i,0}^{(0)} \right) \tag{5}$$

for some $w_{k,i,0}^{(0)}, \dots, w_{k,i,d}^{(0)} \in \mathbb{R}$.

Observe that we have

$$\begin{aligned} T_{\beta_n}(z) &= \max\{-\beta_n, \min\{\beta_n, z\}\} = \max\{0, \beta_n - \max\{-\beta_n, -z\}\} - \beta_n \\ &= \max\{0, 2\beta_n - \max\{0, -z + \beta_n\}\} - \beta_n = \sigma(2\beta_n - \sigma((-1) \cdot z + \beta_n)) - \beta_n, \end{aligned}$$

hence $z \mapsto T_{\beta_n}(z)$ is a neural network with two layers, one hidden neuron per layer and ReLU activation function. This implies that (3) is a neural network which consists of K_n fully connected neural networks of depth L_n+2 and width r in layers $1, \dots, L_n$ and width 1 in layers L_n and $L_n + 1$ computed in parallel and which computes a linear combination of the outputs of these K_n neural networks. The weights in the k-th such network are denoted by $(w_{k,i,j}^{(l)})_{i,j,l}$, where $w_{k,i,j}^{(l)}$ is the weight between neuron j in layer l and neuron i in layer $l + 1$. Here the weights in layers $L_n + 1$ and layer $L_n + 2$ are fixed and the weights in layers $0, \dots, L_n - 1$ and the weights in the output layer are variable.

2.2 Initialization of the weights

We initialize the weights $\mathbf{w}^{(0)} = ((\mathbf{w}^{(0)})_{k,i,j}^{(l))})_{k,i,j,l}$ as follows: We set

$$w_{1,1,j}^{(L_n)} = 0 \quad (j = 1, \dots, K_n)$$

and choose all other $w_{k,i,j}^{(l)}$ $0 \leq l \leq L$ independently from some random distribution as follows: For $1 \leq l < L_n$ we choose

$$w_{k,i,j}^{(l)} \sim U\,[-1, 1]$$

and we choose

$$w_{k,i,j}^{(0)} \sim U[-n, n].$$

2.3 Gradient descent

After initialization of the weights we perform $t_n \in \mathbb{N}$ gradient descent steps with a step size $\lambda_n > 0$. Here we try to minimize the empirical L_2 risk

$$F_n(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^{n} |Y_i - f_{\mathbf{w}}(X_i)|^2. \tag{6}$$

To do this we apply gradient descent together with a projection step which helps us to control the complexity of our over-parametrized deep neural network estimate. More precisely, we choose $\lambda_n, \gamma_n, \delta_n > 0$ (the exact values of λ_n, γ_n and δ_n will be specified in Theorem 1 below), let A be the set of all weight vectors $(w_{1,1,j}^{(L_n)})_{j=1,\dots,K_n}$ which satisfy

$$\sum_{j=1}^{K_n} |w_{1,1,j}^{(L_n)}| \leq \gamma_n,$$

let B be the subsets of all weight vectors $(w^{(l)}_{k,i,j})_{k,i,j,l:l<L_n}$ which satisfy

$$\|(w^{(l)}_{k,i,j})_{k,i,j,l:l<L_n} - ((\mathbf{w}^{(0)})^{(l)}_{k,i,j})_{k,i,j,l:l<L_n}\| \leq \delta_n,$$

and set

$$\left((\mathbf{w}^{(t+1)})^{(L_n)}_{1,1,k}\right)_{k=1,\dots,K_n} = Proj_A\left(\left((\mathbf{w}^{(t)})^{(L_n)}_{1,1,k} - \lambda_n \cdot \frac{\partial F_n(\mathbf{w}^{(t)})}{\partial w^{(L_n)}_{1,1,k}}\right)_{k=1,\dots,K_n}\right)$$

and

$$((\mathbf{w}^{(t+1)})^{(l)}_{k,i,j})_{k,i,j,l:l<L_n} = Proj_B\left(\left((\mathbf{w}^{(t)})^{(l)}_{k,i,j} - \lambda_n \cdot \frac{\partial F_n(\mathbf{w}^{(t)})}{\partial w^{(l)}_{k,i,j}}\right)_{k,i,j,l:l<L_n}\right)$$

for $l < L$. Here

$$\frac{\partial F_n(\mathbf{w}^{(t)})}{\partial w^{(l)}_{k,i,j}} = \begin{cases} \frac{2}{n}\sum_{\rho=1}^n (f_{\mathbf{w}^{(t)}}(X_\rho) - Y_\rho) \cdot T_{\beta_n}(f^{(L_n)}_{\mathbf{w}^{(t)},j,1}(X_\rho)), & \text{if } l = L_n, \\ & k = i = 1 \\ \frac{2}{n}\sum_{\rho=1}^n (f_{\mathbf{w}^{(t)}}(X_\rho) - Y_\rho) \cdot (\mathbf{w}^{(t)})^{(L_n)}_{1,1,k} \cdot \frac{\partial T_{\beta_n}(f^{(L_n)}_{\mathbf{w}^{(t)},k,1}(X_\rho))}{\partial w^{(l)}_{k,i,j}}, & \text{if } l < L_n. \end{cases}$$

The chain rule implies

$$\begin{aligned} \frac{\partial T_{\beta_n}(f^{(L_n)}_{\mathbf{w}^{(t)},k,1}(X_\rho))}{\partial w^{(l)}_{k,i,j}} &= \sigma'\left((-1)\cdot\sigma((-1)\cdot f^{(L_n)}_{\mathbf{w}^{(t)},k,1}(X_\rho) + \beta_n)\right) \cdot \\ & \quad (-1)\cdot\sigma'\left((-1)\cdot f^{(L_n)}_{\mathbf{w}^{(t)},k,1}(X_\rho) + \beta_n\right)\cdot(-1)\cdot\frac{\partial f^{(L_n)}_{\mathbf{w}^{(t)},k,1}(X_\rho)}{\partial w^{(l)}_{k,i,j}} \end{aligned}$$

and

$$\begin{aligned} & \frac{\partial f^{(L_n)}_{\mathbf{w}^{(t)},k,1}(X_\rho)}{\partial w^{(l)}_{k,i,j}} \\ &= \sum_{s_{l+2}=1}^r \cdots \sum_{s_{L-1}=1}^r f^{(l)}_{\mathbf{w}^{(t)},k,j}(x)\cdot\sigma'\left(\sum_{s=1}^r (\mathbf{w}^{(t)})^{(l)}_{k,i,s}\cdot f^{(l)}_{\mathbf{w}^{(t)},k,s}(X_\rho) + (\mathbf{w}^{(t)})^{(l)}_{k,i,0}\right) \\ & \quad \cdot(\mathbf{w}^{(t)})^{(l+1)}_{k,s_{l+2},i}\cdot\sigma'\left(\sum_{s=1}^r (\mathbf{w}^{(t)})^{(l+1)}_{k,s_{l+2},s}\cdot f^{(l+1)}_{\mathbf{w}^{(t)},k,s}(X_\rho) + (\mathbf{w}^{(t)})^{(l+1)}_{k,s_{l+2},0}\right)\cdot(\mathbf{w}^{(t)})^{(l+2)}_{k,s_{l+3},s_{l+2}} \\ & \quad \cdot\sigma'\left(\sum_{s=1}^r (\mathbf{w}^{(t)})^{(l+2)}_{k,s_{l+3},s}\cdot f^{(l+2)}_{\mathbf{w}^{(t)},k,s}(X_\rho) + (\mathbf{w}^{(t)})^{(l+2)}_{k,s_{l+3},0}\right)\cdots(\mathbf{w}^{(t)})^{(L-2)}_{k,s_{L-1},s_{L-2}} \\ & \quad \cdot\sigma'\left(\sum_{s=1}^r (\mathbf{w}^{(t)})^{(L-2)}_{k,s_{L-1},s}\cdot f^{(L-2)}_{\mathbf{w}^{(t)},k,t}(X_\rho) + (\mathbf{w}^{(t)})^{(L-2)}_{k,s_{L-1},0}\right)\cdot(\mathbf{w}^{(t)})^{(L-1)}_{k,1,s_{L-1}} \end{aligned}$$

$$\cdot\sigma'\left(\sum_{s=1}^{r}(\mathbf{w}^{(t)})_{k,1,s}^{(L-1)}\cdot f_{\mathbf{w}^{(t)},k,s}^{(L-1)}(X_\rho)+(\mathbf{w}^{(t)})_{k,1,0}^{(L-1)}\right),$$

where we have used the abbreviations

$$f_{\mathbf{w}^{(t)},k,j}^{(0)}(x)=\begin{cases} x^{(j)} & \text{if } j\in\{1,\dots,d\}\\ 1 & \text{if } j=0\end{cases}$$

and

$$f_{\mathbf{w}^{(t)},k,0}^{(l)}(x)=1 \quad (l=1,\dots,L-1).$$

The ReLU activation function $\sigma(x)=\max\{x,0\}$ is not differentiable at $x=0$, so the above gradient descent procedure is not well defined as soon as one of the arguments of σ' becomes 0. In the sequel we use a subgradient of the convex function σ at zero and set

$$\sigma'(0)=0.$$

Observe that this is relevant only for derivatives with respect to the inner weights of the network, because in case of outer weights $\sigma'(0)$ can not occur.

2.4 Definition of the estimate

We define our estimate as a truncated version of the neural network with weight vector $\mathbf{w}^{(\hat{t})}$ where $\hat{t}\in\{0,1,\dots,t_n\}$ is the index for which the empirical L_2 risk is minimal during the training, i.e., we set

$$\hat{t}=\arg\min_{t\in\{0,1,\dots,t_n\}}\frac{1}{n}\sum_{i=1}^{n}|f_{\mathbf{w}^{(t)}}(X_i)-Y_i|^2 \tag{7}$$

and

$$m_n(x)=T_{\beta_n}(f_{\mathbf{w}^{(\hat{t})}}(x)), \tag{8}$$

where $\beta_n=c_1\cdot\log n$.

3 Main results

Our first result is the following bound on the expected L_2 error of our deep neural network estimate defined in Section 2.

Theorem 1 *Let $p=q+s$ for some $q\in\mathbb{N}_0$ and $s\in(0,1]$, and let $C>0$. Let $n\in\mathbb{N}$, let (X,Y), (X_1,Y_n), ..., (X_n,Y_n) be independent and identically distributed $\mathbb{R}^d\times\mathbb{R}$–valued random variables such that supp$(X)\subseteq[0,1]^d$ and*

$$\mathbf{E}\left\{e^{c_2\cdot Y^2}\right\}<\infty \tag{9}$$

hold and that the corresponding regression function $m(x)=\mathbf{E}\{Y|X=x\}$ is (p,C)-smooth.

Let $\sigma(x) = \max\{x, 0\}$ be the ReLU activation function, choose $K_n \in \mathbb{N}$ such that

$$\frac{K_n}{n^{c_3 \cdot ((\log n)+1)}} \to \infty \quad (n \to \infty) \tag{10}$$

holds for c_3 sufficiently large. Choose c_4, c_5 and c_6 sufficiently large, set

$$\delta_n = 1, \gamma_n = c_4 \cdot n^{\frac{d}{2\cdot(2p+d)}}, \quad L_n = c_5 \cdot \log n, \quad r = c_6,$$

$$t_n = K_n^2 \cdot n^2, \quad \text{and} \quad \lambda_n = \frac{1}{t_n}.$$

and define the estimate m_n as in Section 2. Then we have for n sufficiently large

$$\mathbf{E} \int |m_n(x) - m(x)|^2 \mathbf{P}_X(dx) \leq c_8 \cdot (\log n)^5 \cdot n^{-\frac{p}{2p+d}}.$$

Remark 1. According to Stone (1982) the above rate of convergence is not optimal. We see our result nevertheless as extremely useful because it introduces a new proof technique which enables to analyze the rate of convergence of over-parametrized deep neural network estimates with ReLU activation function learned by gradient descent. We believe that it is necessary to improve the bound on the generalization error in Lemma 5 below in order to improve the above rate of convergence result.

Remark 2. It follows from the proof of Theorem 1 below, that the result also holds if the gradient descent is used only for the outer weights of the network. In this sense our result indicates that the random initialization of the inner weights together with the over-parametrization is important, not the gradient descent.

The rate of convergence in the above theorem gets worse if the dimension of X gets large. This is due to the well–known curse of dimensionality. In the sequel we show that by adapting the parameter γ_n of our estimate, we can get a better rate of convergence if the regression functions satisfies the assumption of an interaction model. Here it is assumed that

$$m(x) = \sum_{I \subseteq \{1,\dots,d\} \,:\, |I| = d^*} m_I(x_I),$$

where $1 \leq d^* < d$, $m_I : \mathbb{R}^{d^*} \to \mathbb{R}$ ($I \subseteq \{1, \dots, d\}$, $|I| = d^*$) are (p, C)-smooth functions and we have used the notation

$$x_I = (x^{(j_1)}, \dots, x^{(j_{d^*})})$$

for $I = \{j_1, \dots, j_{d^*}\}$.

Theorem 2 *Let $p = q + s$ for some $q \in \mathbb{N}_0$ and $s \in (0, 1]$, and let $C > 0$. Let $n \in \mathbb{N}$, let (X, Y), (X_1, Y_n), ..., (X_n, Y_n) be independent and identically distributed $\mathbb{R}^d \times \mathbb{R}$–valued* *random variables such that supp(X)* $\subseteq [0,1]^d$ *and (9) hold and that the corresponding regression function* $m(x) = \mathbf{E}\{Y|X=x\}$ *satisfies*

$$m(x) = \sum_{I \subseteq \{1,\dots,d\} \,:\, |I| = d^*} m_I(x_I)$$

for some (p,C)*–smooth function* $m_I : [0,1]^{d^*} \to \mathbb{R}$*. Define the estimate as in Theorem 1, except that this time the value of* γ_n *is given by*

$$\gamma_n = c_4 \cdot n^{\frac{d^*}{2\cdot(2p+d^*)}}.$$

Then we have for n *sufficiently large*

$$\mathbf{E}\int |m_n(x) - m(x)|^2 \mathbf{P}_X(dx) \leq c_8 \cdot (\log n)^5 \cdot n^{-\frac{p}{2p+d^*}}.$$

Remark 3. The rate of convergence derived in Theorem 2 does not depend on d, hence under the above assumption on the regression function our estimate is able to circumvent the curse of dimensionality. That this is possible is well-known (cf., Stone (1994) and the literature cited therein), in particular Stone (1994) shows that our rate of convergence in Theorem 2 is not optimal. As in Theorem 1 we believe that this is not due to the estimate but due to our proof. However, we would like to stress that our result is the first result which shows that (over-parametrized) neural network estimates with ReLU activation function learned by the gradient descent can achieve a dimension-free rate of convergence in case of interaction models.

4 Proofs

4.1 Optimization error

Lemma 1 *Let* $d_1, d_2 \in \mathbb{N}$*, let* $C_n, D_n \geq 0$*, let* $A \subset \mathbb{R}^{d_1}$ *and* $B \subseteq \mathbb{R}^{d_2}$ *be closed and convex, and let* $F : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to \mathbb{R}_+$ *be a function such that*

$$u \mapsto F(u,v) \quad \textit{is differentiable and convex for all } v \in \mathbb{R}^{d_1}$$

and

$$\|(\nabla_u F)(u,v)\| \leq C_n \tag{11}$$

for all $(u,v) \in A \times B$*. Choose* $(u_0, v_0) \in A \times B$*, let* $v_1, \dots, v_{t_n} \in B$ *and set*

$$u_{t+1} = Proj_A\left(u_t - \lambda \cdot (\nabla_u F)(u_t, v_t)\right),$$

where

$$\lambda = \frac{1}{t_n}$$

Let $u^* \in A$ *and assume*

$$|F(u^*, v_t) - F(u^*, v_1)| \leq D_n \cdot \|u^*\| \cdot \|v_t - v_1\| \tag{12}$$

for all $t = 1, \ldots, t_n$*. Then it holds:*

$$\min_{t=0,\ldots,t_n} F(u_t, v_t) \leq F(u^*, v_0) + D_n \cdot \|u^*\| \cdot diam(B) + \frac{\|u^* - u_0\|^2}{2} + \frac{C_n^2}{2 \cdot t_n}.$$

Proof. In the *first step of the proof* we show

$$\frac{1}{t_n} \sum_{t=0}^{t_n-1} F(u_t, v_t) \leq \frac{1}{t_n} \sum_{t=0}^{t_n-1} F(u^*, v_t) + \frac{\|u^* - u_0\|^2}{2} + \frac{1}{2 \cdot t_n^2} \sum_{t=0}^{t_n-1} \|(\nabla_u F)(u_t, v_t)\|^2. \quad (13)$$

By convexity of $u \mapsto F(u, v_t)$ and because of $u^* \in A$ we have

$$\begin{aligned}
& F(u_t, v_t) - F(u^*, v_t) \\
& \leq < (\nabla_u F)(u_t, v_t), u_t - u^* > \\
& = \frac{1}{2 \cdot \lambda} \cdot 2 \cdot < \lambda \cdot (\nabla_u F)(u_t, v_t), u_t - u^* > \\
& = \frac{1}{2 \cdot \lambda} \cdot \left(-\|u_t - u^* - \lambda \cdot (\nabla_u F)(u_t, v_t)\|^2 + \|u_t - u^*\|^2 + \|\lambda \cdot (\nabla_u F)(u_t, v_t)\|^2 \right) \\
& \leq \frac{1}{2 \cdot \lambda} \cdot \left(-\|Proj_A(u_t - \lambda \cdot (\nabla_u F)(u_t, v_t)) - u^*\|^2 + \|u_t - u^*\|^2 + \lambda^2 \cdot \|(\nabla_u F)(u_t, v_t)\|^2 \right) \\
& = \frac{1}{2 \cdot \lambda} \cdot \left(\|u_t - u^*\|^2 - \|u_{t+1} - u^*\|^2 + \lambda^2 \cdot \|(\nabla_u F)(u_t, v_t)\|^2 \right).
\end{aligned}$$

This implies

$$\begin{aligned}
& \frac{1}{t_n} \sum_{t=0}^{t_n-1} F(u_t, v_t) - \frac{1}{t_n} \sum_{t=0}^{t_n-1} F(u^*, v_t) \\
& = \frac{1}{t_n} \sum_{t=0}^{t_n-1} \left(F(u_t, v_t) - F(u^*, v_t) \right) \\
& \leq \frac{1}{t_n} \sum_{t=0}^{t_n-1} \frac{1}{2 \cdot \lambda} \cdot \left(\|u_t - u^*\|^2 - \|u_{t+1} - u^*\|^2 \right) + \frac{1}{t_n} \sum_{t=0}^{t_n-1} \frac{\lambda}{2} \cdot \|(\nabla_u F)(u_t, v_t)\|^2 \\
& = \frac{1}{2} \cdot \sum_{t=0}^{t_n-1} \left(\|u_t - u^*\|^2 - \|u_{t+1} - u^*\|^2 \right) + \frac{1}{2 \cdot t_n^2} \sum_{t=0}^{t_n-1} \|(\nabla_u F)(u_t, v_t)\|^2 \\
& \leq \frac{\|u_0 - u^*\|^2}{2} + \frac{1}{2 \cdot t_n^2} \sum_{t=0}^{t_n-1} \|(\nabla_u F)(u_t, v_t)\|^2.
\end{aligned}$$

In the *second step of the proof* we show the assertion.
Using the result of step 1 we get

$$\begin{aligned}
& \min_{t=0,\ldots,t_n} F(u_t, v_t) \\
& \leq \frac{1}{t_n} \sum_{t=0}^{t_n-1} F(u_t, v_t)
\end{aligned}$$

$$\le \frac{1}{t_n}\sum_{t=0}^{t_n-1} F(u^*,v_t) + \frac{\|u^*-u_0\|^2}{2} + \frac{1}{2\cdot t_n^2}\sum_{t=0}^{t_n-1}\|(\nabla_u F)(u_t,v_t)\|^2$$
$$\le F(u^*,v_0) + \frac{1}{t_n}\sum_{t=0}^{t_n-1}|F(u^*,v_t)-F(u^*,v_0)| + \frac{\|u^*-u_0\|^2}{2}$$
$$+\frac{1}{2\cdot t_n^2}\sum_{t=0}^{t_n-1}\|(\nabla_u F)(u_t,v_t)\|^2.$$

By (12) we get

$$\begin{aligned}\frac{1}{t_n}\sum_{t=0}^{t_n-1}|F(u^*,v_t)-F(u^*,v_0)| &\le \frac{1}{t_n}\sum_{t=0}^{t_n-1} D_n\cdot\|u^*\|\cdot\|v_t-v_0\| \\ &\le D_n\cdot\|u^*\|\cdot diam(B).\end{aligned}$$

And by (11) we get

$$\frac{1}{2\cdot t_n^2}\sum_{t=0}^{t_n-1}\|(\nabla_u F)(u_t,v_t)\|^2 \le \frac{1}{2\cdot t_n^2}\sum_{t=0}^{t_n-1} C_n^2 = \frac{C_n^2}{2\cdot t_n}.$$

Summarizing the above results, the proof is complete. □

Lemma 2 *Let $\sigma:\mathbb{R}\to\mathbb{R}$ be arbitrary and define $F_n(\mathbf{w})$ by (6). Then we have*

$$\|(\nabla_{(w_{1,1,k}^{(L_n)})_{k=1,\dots,K_n}} F_n)(\mathbf{w})\| \le 4\cdot K_n\cdot\beta_n^2\cdot F_n(\mathbf{w}).$$

Proof. We have

$$\begin{aligned}&\|(\nabla_{(w_{1,1,k}^{(L_n)})_{k=1,\dots,K_n}} F_n)(\mathbf{w})\|^2\\ &=\sum_{k=1}^{K_n}\left|\frac{2}{n}\sum_{\rho=1}^{n}(f_{\mathbf{w}}(X_\rho)-Y_\rho)\cdot T_{\beta_n}(f_{\mathbf{w},k,1}^{(L_n)}(X_\rho))\right|^2\\ &\le\sum_{k=1}^{K_n}\left(4\cdot\frac{1}{n}\sum_{\rho=1}^{n}|f_{\mathbf{w}}(X_\rho)-Y_\rho|^2\cdot\frac{1}{n}\sum_{\rho=1}^{n}(T_{\beta_n}(f_{\mathbf{w},k,1}^{(L_n)}(X_\rho)))^2\right)\\ &\le 4\cdot K_n\cdot F_n(\mathbf{w})\cdot\beta_n^2.\end{aligned}$$

□

Lemma 3 *Let $\sigma:\mathbb{R}\to\mathbb{R}$ be arbitrary, let $\gamma_n\ge 1$ and define $F_n(\mathbf{w})$ by (6). Assume $Y_1,\dots,Y_n\in[-\beta_n,\beta_n]$ and*

$$\sum_{k=1}^{K_n}|w_{1,1,k}^{(L_n)}|\le\gamma_n. \qquad (14)$$

Then we have

$$F_n(\mathbf{w})\le(2+2\cdot\gamma_n^2)\cdot\beta_n^2.$$

Proof. Using (14) we get

$$F_n(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^n |Y_i - f_{\mathbf{w}}(X_i)|^2 \leq \frac{1}{n}\sum_{i=1}^n (2 \cdot Y_i^2 + 2 \cdot |\gamma_n \cdot \beta_n|^2) \leq (2 + 2 \cdot \gamma_n^2) \cdot \beta_n^2.$$

□

Lemma 4 *Let $\sigma : \mathbb{R} \to \mathbb{R}$ be the ReLU activation function, let $A, \gamma_n \geq 1$, let $\delta_n \geq 0$ and define $f_{\mathbf{w},1,1}^{(L_n)}(x)$ be (4) and (5). Assume*

$$\sum_{j=0}^r (w_{1,i,j}^{(l)})_+ \leq 1 + \delta_n \quad \textit{and} \quad \sum_{j=0}^r (w_{1,i,j}^{(l)})_- \leq 1 + \delta_n \quad (l \in \{1, \dots, L_n - 1\}), \tag{15}$$

$$|w_{1,i,j}^{(0)}| \leq A \tag{16}$$

and

$$\|(w_{1,i,j}^{(l)})_{i,j,l\,:\,l<L_n} - (\bar{w}_{1,i,j}^{(l)})_{i,j,l\,:\,l<L_n}\|_\infty \leq 1. \tag{17}$$

Then we have for any $\bar{\mathbf{w}}$ and any $x \in [0,1]^d$:

$$\begin{aligned}&|f_{\mathbf{w},1,1}^{(L_n)}(x) - f_{\bar{\mathbf{w}},1,1}^{(L_n)}(x)| \\ &\leq (4 \cdot r)^{L_n} \cdot (d+1) \cdot A \cdot (1+\delta_n)^{L_n - 1} \cdot \|(w_{1,i,j}^{(l)})_{i,j,l\,:\,l<L_n} - (\bar{w}_{1,i,j}^{(l)})_{i,j,l\,:\,l<L_n}\|_\infty.\end{aligned}$$

Proof. In the *first step* of the proof we show

$$0 \leq f_{\mathbf{w},1,1}^{(l)}(x) \leq (d+1) \cdot A \cdot (1+\delta_n)^{l-1}$$

for all $l = 1, \dots, L_n$.

Using $0 \leq \sigma(x)| \leq |x|$ and (16) we get

$$0 \leq f_{\mathbf{w},1,i}^{(1)}(x) \leq \sum_{j=1}^d |w_{1,i,j}^{(0)}| \cdot |x^{(j)}| + |w_{1,i,0}^{(0)}| \leq (d+1) \cdot A \cdot (1+\delta_n)^{1-1}.$$

Recursively we can conclude from (15)

$$\begin{aligned}0 \leq f_{\mathbf{w},1,i}^{(l)}(x) &\leq \left| \sum_{j=0}^r (w_{1,i,j}^{(l-1)})_+ \cdot f_{1,j}^{(l-1)}(x) - \sum_{j=0}^r (w_{1,i,j}^{(l-1)})_- \cdot f_{1,j}^{(l-1)}(x) \right| \\ &\leq (1+\delta_n) \cdot (d+1) \cdot A \cdot (1+\delta_n)^{l-2} = (d+1) \cdot A \cdot (1+\delta_n)^{l-1}\end{aligned}$$

for $l = 2, \dots, L_n$, where we have set $f_{1,0}^{(l-1)}(x) = 1$.

In the *second step* of the proof we show

$$\begin{aligned}&|f_{\mathbf{w},1,k}^{(l)}(x) - f_{\bar{\mathbf{w}},1,k}^{(l)}(x)| \\ &\leq (4 \cdot r)^l \cdot (d+1) \cdot A \cdot (1+\delta_n)^{l-1} \cdot \|(w_{1,i,j}^{(l)})_{i,j,l\,:\,l<L_n} - (\bar{w}_{1,i,j}^{(l)})_{i,j,l\,:\,l<L_n}\|_\infty\end{aligned} \tag{18}$$

for $l = 1, \dots, L_n$.

The ReLu activation function is Lipschitz continuous with Lipschitz constant 1. Consequently we get for $l = 1$

$$\begin{aligned} |f^{(1)}_{\mathbf{w},1,i}(x) - f^{(1)}_{\bar{\mathbf{w}},1,i}(x)| &\leq \sum_{j=1}^{d} |w^{(0)}_{1,i,j} - \bar{w}^{(0)}_{1,i,j}| \cdot |x^{(j)}| + |w^{(0)}_{1,i,0} - \bar{w}^{(0)}_{1,i,0}| \\ &\leq (d+1) \cdot \|(w^{(l)}_{1,i,j})_{i,j,l\,:\,l<L_n} - (\bar{w}^{(l)}_{1,i,j})_{i,j,l\,:\,l<L_n}\|_\infty. \end{aligned}$$

Assume now that (18) holds for some $l \in \{1, \dots, L_n - 1\}$. Then we can conclude from the first step of the proof

$$\begin{aligned} &|f^{(l+1)}_{\mathbf{w},1,i}(x) - f^{(l+1)}_{\bar{\mathbf{w}},1,i}(x)| \\ &= \left| \sigma\left(\sum_{j=1}^{r} w^{(l)}_{1,i,j} \cdot f^{(l)}_{\mathbf{w},1,j}(x) + w^{(l)}_{1,i,0} \right) - \sigma\left(\sum_{j=1}^{r} \bar{w}^{(l)}_{1,i,j} \cdot f^{(l)}_{\bar{\mathbf{w}},1,j}(x) + \bar{w}^{(l)}_{1,i,0} \right) \right| \\ &\leq \left| \sum_{j=1}^{r} w^{(l)}_{1,i,j} \cdot f^{(l)}_{\mathbf{w},1,j}(x) + w^{(l)}_{1,i,0} - \sum_{j=1}^{r} \bar{w}^{(l)}_{1,i,j} \cdot f^{(l)}_{\bar{\mathbf{w}},1,j}(x) - \bar{w}^{(l)}_{1,i,0} \right| \\ &\leq \sum_{j=1}^{r} |w^{(l)}_{1,i,j} - \bar{w}^{(l)}_{1,i,j}| \cdot |f^{(l)}_{\mathbf{w},1,j}(x)| + |w^{(l)}_{1,i,0} - \bar{w}^{(l)}_{1,i,0}| \\ &\quad + \sum_{j=1}^{r} |\bar{w}^{(l)}_{1,i,j}| \cdot |f^{(l)}_{\mathbf{w},k,j}(x) - f^{(l)}_{\bar{\mathbf{w}},1,j}(x)| \\ &\leq (r \cdot (d+1) \cdot A \cdot (1+\delta_n)^{l-1} + 1) \cdot \|(w^{(l)}_{1,i,j})_{i,j,l\,:\,l<L_n} - (\bar{w}^{(l)}_{1,i,j})_{i,j,l\,:\,l<L_n}\|_\infty \\ &\quad + r \cdot \Big(2 \cdot (1+\delta_n) + \|(w^{(l)}_{1,i,j})_{i,j,l\,:\,l<L_n} - (\bar{w}^{(l)}_{1,i,j})_{i,j,l\,:\,l<L_n}\|_\infty \Big) \cdot (4 \cdot r)^l \\ &\qquad\qquad \cdot (d+1) \cdot A \cdot (1+\delta_n)^{l-1} \cdot \|(w^{(l)}_{1,i,j})_{i,j,l\,:\,l<L_n} - (\bar{w}^{(l)}_{1,i,j})_{i,j,l\,:\,l<L_n}\|_\infty \\ &\leq (4 \cdot r)^{l+1} \cdot (d+1) \cdot A \cdot (1+\delta_n)^l \cdot \|(w^{(l)}_{1,i,j})_{i,j,l\,:\,l<L_n} - (\bar{w}^{(l)}_{1,i,j})_{i,j,l\,:\,l<L_n}\|_\infty. \end{aligned}$$

□

4.2 Generalization error

Lemma 5 *Let $L_n, K_n \in \mathbb{N}$ and $\gamma_n > 0$. Assume $L_n \leq n^{c_9}$. Let $\sigma(x) = \max\{x, 0\}$ be the ReLU activation function. Let $\mathcal{F}$ be the set of all functions $f_{\mathbf{w}}$ defined by (3)–(5) where the weight vector $\mathbf{w}$ satisfies*

$$\sum_{j=1}^{K_n} |w^{(L_n)}_{1,1,j}| \leq \gamma_n. \tag{19}$$

Let (X,Y), (X_1,Y_1), $\ldots$, (X_n,Y_n) be independent and identically distributed $[0,1]^d \times [-\beta_n,\beta_n]$*-valued random vectors. Then we have*

$$\mathbf{E}\left\{\sup_{f\in\mathcal{F}}\left(\mathbf{E}\{|(T_{\beta_n}f)(X)-Y|^2\}-\frac{1}{n}\sum_{i=1}^n|(T_{\beta_n}f)(X_i)-Y_i|^2\right)\right\}$$
$$\leq c_{10}\cdot\frac{\gamma_n\cdot L_n\cdot\sqrt{\log L_n}\cdot\beta_n\cdot\log n}{\sqrt{n}}.$$

Proof. In the *first step of the proof* we show

$$\mathbf{E}\left\{\sup_{f\in\mathcal{F}}\left(\mathbf{E}\{|(T_{\beta_n}f)(X)-Y|^2\}-\frac{1}{n}\sum_{i=1}^n|(T_{\beta_n}f)(X_i)-Y_i|^2\right)\right\}$$
$$\leq 8\cdot\beta_n\cdot\mathbf{E}\left\{\sup_{f\in\mathcal{F}}\frac{1}{n}\sum_{i=1}^n\epsilon_i\cdot f(X_i)\right\}, \qquad (20)$$

where $\epsilon_1,\ldots,\epsilon_n$ are random variables satisfying $\mathbf{P}\{\epsilon_j=1\}=\mathbf{P}\{\epsilon_j=-1\}=1/2$ $(j=1,\ldots,n)$ and $X_1,\ldots,X_n,\epsilon_1,\ldots,\epsilon_n$ independent.

Choose random variables (X_1',Y_1'), $\ldots$, (X_n',Y_n') such that

$$(X_1,Y_1),\ldots,(X_n,Y_n),\epsilon_1,\ldots,\epsilon_n,(X_1',Y_1'),\ldots,(X_n',Y_n')$$

are independent and such that

$$(X_1,Y_1),\ldots,(X_n,Y_n),(X_1',Y_1'),\ldots,(X_n',Y_n')$$

are identically distributed and set $(X,Y)_1^n=((X_1,Y_1),\ldots,(X_n,Y_n))$. We have

$$\mathbf{E}\left\{\sup_{f\in\mathcal{F}}\left(\mathbf{E}\{|(T_{\beta_n}f)(X)-Y|^2\}-\frac{1}{n}\sum_{i=1}^n|(T_{\beta_n}f)(X_i)-Y_i|^2\right)\right\}$$
$$=\mathbf{E}\left\{\sup_{f\in\mathcal{F}}\left(\mathbf{E}\{\frac{1}{n}\sum_{i=1}^n|(T_{\beta_n}f)(X_i')-Y_i'|^2|(X,Y)_1^n\}-\frac{1}{n}\sum_{i=1}^n|(T_{\beta_n}f)(X_i)-Y_i|^2\right)\right\}$$
$$\leq\mathbf{E}\left\{\mathbf{E}\left\{\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^n|(T_{\beta_n}f)(X_i')-Y_i'|^2-\frac{1}{n}\sum_{i=1}^n|(T_{\beta_n}f)(X_i)-Y_i|^2\right)|(X,Y)_1^n\right\}\right\}$$
$$=\mathbf{E}\left\{\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^n|(T_{\beta_n}f)(X_i')-Y_i'|^2-\frac{1}{n}\sum_{i=1}^n|(T_{\beta_n}f)(X_i)-Y_i|^2\right)\right\}.$$

Since the joint distribution of $(X_1,Y_1),\ldots,(X_n,Y_n),(X_1',Y_1'),\ldots,(X_n',Y_n')$ does not change if we (randomly) interchange (X_i,Y_i) and (X_i',Y_i'), the last term is equal to

$$\mathbf{E}\left\{\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^n\epsilon_i\cdot\left(|(T_{\beta_n}f)(X_i')-Y_i'|^2-|(T_{\beta_n}f)(X_i)-Y_i|^2\right)\right)\right\}$$

$$\leq \mathbf{E}\left\{\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\cdot|(T_{\beta_n}f)(X_i')-Y_i'|^2\right)\right\}+\mathbf{E}\left\{\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^{n}(-\epsilon_i)\cdot|(T_{\beta_n}f)(X_i)-Y_i|^2\right)\right\}$$
$$= 2\cdot\mathbf{E}\left\{\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\cdot|(T_{\beta_n}f)(X_i)-Y_i|^2\right)\right\}.$$

Next we use a contraction style argument. Because of the independence of the random variables we can compute the expectation by first computing the expectation with respect to ϵ_1 and then computing the expectation with respect to all other random variables. This yields that the last term above is equal to

$$2\cdot\mathbf{E}\Bigg\{\frac{1}{2}\cdot\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}f)(X_i)-Y_i|^2+\frac{1}{n}\cdot|(T_{\beta_n}f)(X_1)-Y_1|^2\right)$$
$$+\frac{1}{2}\cdot\sup_{g\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}g)(X_i)-Y_i|^2\right)-\frac{1}{n}\cdot|(T_{\beta_n}g)(X_1)-Y_1|^2\Bigg\}$$
$$=\mathbf{E}\Bigg\{\sup_{f,g\in\mathcal{F}}\Bigg(\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}f)(X_i)-Y_i|^2+\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}g)(X_i)-Y_i|^2$$
$$+\frac{1}{n}\cdot|(T_{\beta_n}f)(X_1)-Y_1|^2-\frac{1}{n}\cdot|(T_{\beta_n}g)(X_1)-Y_1|^2\Bigg)\Bigg\}$$
$$\leq\mathbf{E}\Bigg\{\sup_{f,g\in\mathcal{F}}\Bigg(\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}f)(X_i)-Y_i|^2+\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}g)(X_i)-Y_i|^2$$
$$+\frac{4\beta_n}{n}\cdot|(T_{\beta_n}f)(X_1)-(T_{\beta_n}g)(X_1)|\Bigg)\Bigg\}$$
$$\leq\mathbf{E}\Bigg\{\sup_{f,g\in\mathcal{F}}\Bigg(\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}f)(X_i)-Y_i|^2+\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}g)(X_i)-Y_i|^2$$
$$+\frac{4\beta_n}{n}\cdot|f(X_1)-g(X_1)|\Bigg)\Bigg\}.$$

For fixed (X_1,Y_1), $\dots$, (X_n,Y_n), ϵ_2, $\dots$, ϵ_n the term

$$\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}f)(X_i)-Y_i|^2+\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}g)(X_i)-Y_i|^2+\frac{4\beta_n}{n}\cdot|f(X_1)-g(X_1)|$$

is symmetric in f and g. Therefore we can assume w.l.o.g. that $f(X_1)\geq g(X_1)$ holds which implies that we have

$$\sup_{f,g\in\mathcal{F}}\Bigg(\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}f)(X_i)-Y_i|^2+\frac{1}{n}\sum_{i=2}^{n}\epsilon_i\cdot|(T_{\beta_n}g)(X_i)-Y_i|^2$$
$$+\frac{4\beta_n}{n}\cdot|f(X_1)-g(X_1)|\Bigg)$$

$$= \sup_{f,g \in \mathcal{F}} \Bigg(\frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2 + \frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} g)(X_i) - Y_i|^2 \\ + \frac{4\beta_n}{n} \cdot (f(X_1) - g(X_1)) \Bigg).$$

In the same way we see that the term above is also equal to

$$\sup_{f,g \in \mathcal{F}} \Bigg(\frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2 + \frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} g)(X_i) - Y_i|^2 - \frac{4\beta_n}{n} \cdot (f(X_1) - g(X_1)) \Bigg),$$

and we get

$$\begin{aligned}
&\mathbf{E}\Bigg\{ \sup_{f,g \in \mathcal{F}} \Bigg(\frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2 + \frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} g)(X_i) - Y_i|^2 \\
&\qquad + \frac{4\beta_n}{n} \cdot |f(X_1) - g(X_1)| \Bigg) \Bigg\} \\
&= \mathbf{E}\Bigg\{ \frac{1}{2} \cdot \sup_{f,g \in \mathcal{F}} \Bigg(\frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2 + \frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} g)(X_i) - Y_i|^2 \\
&\qquad + \frac{4\beta_n}{n} \cdot (f(X_1) - g(X_1)) \Bigg) \\
&\quad + \frac{1}{2} \cdot \sup_{f,g \in \mathcal{F}} \Bigg(\frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2 + \frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} g)(X_i) - Y_i|^2 \\
&\qquad - \frac{4\beta_n}{n} \cdot (f(X_1) - g(X_1)) \Bigg) \Bigg\} \\
&= \mathbf{E}\Bigg\{ \sup_{f,g \in \mathcal{F}} \Bigg(\frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2 + \frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} g)(X_i) - Y_i|^2 \\
&\qquad + \frac{4\beta_n}{n} \cdot \epsilon_1 \cdot (f(X_1) - g(X_1)) \Bigg) \Bigg\} \\
&\leq \mathbf{E}\Bigg\{ \sup_{f,g \in \mathcal{F}} \Bigg(\frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2 + \frac{4\beta_n}{n} \cdot \epsilon_1 \cdot f(X_1) \Bigg) \Bigg\} \\
&\quad + \sup_{f,g \in \mathcal{F}} \Bigg(\frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} g)(X_i) - Y_i|^2 + \frac{4\beta_n}{n} \cdot (-\epsilon_1) \cdot g(X_1) \Bigg) \Bigg\} \\
&= 2 \cdot \mathbf{E}\Bigg\{ \sup_{f \in \mathcal{F}} \Bigg(\frac{1}{n} \sum_{i=2}^n \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2 + \frac{4\beta_n}{n} \cdot \epsilon_1 \cdot f(X_1) \Bigg) \Bigg\},
\end{aligned}$$

where we have used that $-\epsilon_1$ has the same distribution as ϵ_1.

Arguing in the same way for $i = 2, \dots, n$ we get

$$\begin{aligned}
&2 \cdot \mathbf{E}\left\{\sup_{f\in\mathcal{F}} \left(\frac{1}{n}\sum_{i=1}^{n} \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2\right)\right\} \\
&\leq 2 \cdot \mathbf{E}\left\{\sup_{f\in\mathcal{F}} \left(\frac{1}{n}\sum_{i=2}^{n} \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2 + \frac{4\beta_n}{n} \cdot \epsilon_1 \cdot f(X_1)\right)\right\} \\
&\leq 2 \cdot \mathbf{E}\left\{\sup_{f\in\mathcal{F}} \left(\frac{1}{n}\sum_{i=3}^{n} \epsilon_i \cdot |(T_{\beta_n} f)(X_i) - Y_i|^2 + \frac{4\beta_n}{n} \cdot (\epsilon_1 \cdot f(X_1) + \epsilon_2 \cdot f(X_2))\right)\right\} \\
&\leq \dots \\
&\leq 2 \cdot \mathbf{E}\left\{\sup_{f\in\mathcal{F}} \frac{4\beta_n}{n} \cdot \sum_{i=1}^{n} \epsilon_i \cdot f(X_i)\right\},
\end{aligned}$$

which finishes the first step of the proof.

Let $\mathcal{W}$ be the set of all weight vectors $\mathbf{w} = (w_{i,j,k}^{(l)})_{i,j,k,l}$ which satisfy (19). In the *second step of the proof* we show

$$\mathbf{E}\left\{\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \epsilon_i \cdot f(X_i)\right\} \leq \gamma_n \cdot \mathbf{E}\left\{\sup_{\mathbf{w}\in\mathcal{W}} \left|\frac{1}{n}\sum_{i=1}^{n} \epsilon_i \cdot (T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}(X_i))\right|\right\}.$$

We have

$$\begin{aligned}
&\mathbf{E}\left\{\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \epsilon_i \cdot f(X_i)\right\} \\
&= \mathbf{E}\left\{\sup_{\mathbf{w}\in\mathcal{W}} \frac{1}{n}\sum_{i=1}^{n} \epsilon_i \cdot \sum_{j=1}^{K_n} w_{1,1,j}^{(L_n)} \cdot (T_{\beta_n} f_{\mathbf{w},j,1}^{(L_n)}(X_i)))\right\} \\
&= \mathbf{E}\left\{\sup_{\mathbf{w}\in\mathcal{W}} \sum_{j=1}^{K_n} w_{1,1,j}^{(L_n)} \cdot \frac{1}{n}\sum_{i=1}^{n} \epsilon_i \cdot (T_{\beta_n} f_{\mathbf{w},j,1}^{(L_n)}(X_i)))\right\} \\
&\leq \mathbf{E}\left\{\sup_{\mathbf{w}\in\mathcal{W}} \sum_{j=1}^{K_n} |w_{1,1,j}^{(L_n)}| \cdot \left|\frac{1}{n}\sum_{i=1}^{n} \epsilon_i \cdot (T_{\beta_n} f_{\mathbf{w},j,1}^{(L_n)}(X_i)))\right|\right\} \\
&\leq \mathbf{E}\left\{\sup_{\mathbf{w}\in\mathcal{W}} \sum_{j=1}^{K_n} |w_{1,1,j}^{(L_n)}| \cdot \sup_{\mathbf{w}\in\mathcal{W}, k\in\{1,\dots,K_n\}} \left|\frac{1}{n}\sum_{i=1}^{n} \epsilon_i \cdot (T_{\beta_n} f_{\mathbf{w},k,1}^{(L_n)}(X_i)))\right|\right\} \\
&\leq \gamma_n \cdot \mathbf{E}\left\{\sup_{\mathbf{w}\in\mathcal{W}, k\in\{1,\dots,K_n\}} \left|\frac{1}{n}\sum_{i=1}^{n} \epsilon_i \cdot (T_{\beta_n} f_{\mathbf{w},k,1}^{(L_n)}(X_i)))\right|\right\} \\
&= \gamma_n \cdot \mathbf{E}\left\{\sup_{\mathbf{w}\in\mathcal{W}} \left|\frac{1}{n}\sum_{i=1}^{n} \epsilon_i \cdot (T_{\beta_n} f_{\mathbf{w},1,1}^{(L_n)}(X_i)))\right|\right\},
\end{aligned}$$

where the last inequality followed from

$$\{T_{\beta_n} f_{\mathbf{w},k,1}^{(L_n)} : \mathbf{w} \in \mathcal{W}, k \in \{1, \dots, K_n\}\} = \{T_{\beta_n} f_{\mathbf{w},1,1}^{(L_n)} : \mathbf{w} \in \mathcal{W}\}.$$

In the *third step of the proof* we show

$$\mathbf{E}\left\{\sup_{\mathbf{w}\in\mathcal{W}}\left|\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\cdot(T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}(X_i))\right|\right\}\leq c_{10}\cdot\frac{L_n\cdot\sqrt{\log L_n}\cdot\log n\cdot\beta_n}{\sqrt{n}}.$$

For $\delta_n>0$ we have

$$\begin{aligned}
&\mathbf{E}\left\{\sup_{\mathbf{w}\in\mathcal{W}}\left|\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\cdot(T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}(X_i))\right|\right\}\\
&=\int_0^\infty\mathbf{P}\left\{\sup_{\mathbf{w}\in\mathcal{W}}\left|\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\cdot(T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}(X_i))\right|>t\right\}dt\\
&\leq\delta_n+\int_{\delta_n}^\infty\mathbf{P}\left\{\sup_{\mathbf{w}\in\mathcal{W}}\left|\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\cdot(T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}(X_i))\right|>t\right\}dt.
\end{aligned}$$

Using a standard covering argument from empirical process theory we see that for any $t\geq\delta_n$ we have

$$\begin{aligned}
&\mathbf{P}\left\{\sup_{\mathbf{w}\in\mathcal{W}}\left|\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\cdot(T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}(X_i))\right|>t\right\}\\
&\leq\sup_{x_1^n\in[0,1]^d}\mathcal{M}_1\left(\frac{\delta_n}{2},\left\{T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}):\mathbf{w}\in\mathcal{W}\right\},x_1^n\right)\\
&\qquad\qquad\cdot\sup_{\mathbf{w}\in\mathcal{W}}\mathbf{P}\left\{\left|\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\cdot(T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}(X_i))\right|>\frac{t}{2}\right\}.
\end{aligned}$$

$f_{\mathbf{w},1,1}^{(L_n)}$ is a neural network with at most $c_{11}\cdot r^2\cdot L_n$ many weights and depth L_n+2. Application of Theorem 6 in Bartlett et al. (2019) and Theorem 9.4 in Györfi et al. (2002) yields

$$\sup_{x_1^n\in[0,1]^d}\mathcal{M}_1\left(\frac{\delta_n}{2},\left\{T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}):\mathbf{w}\in\mathcal{W}\right\},x_1^n\right)\leq c_{12}\cdot\left(\frac{c_{13}\cdot\beta_n}{\delta_n}\right)^{c_{14}\cdot L_n^2\cdot r^2\cdot\log(L_n)}.$$

By the inequality of Hoeffding (cf., e.g., Lemma A.3 in Györfi et al. (2002)) and

$$|T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}(x)|\leq\beta_n\quad(x\in\mathbb{R}^d)$$

it holds for any $\mathbf{w}\in\mathcal{W}$

$$\mathbf{P}\left\{\left|\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\cdot T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}(X_i))\right|>t\right\}\leq2\cdot\exp\left(-\frac{2\cdot n\cdot t^2}{4\cdot\beta_n^2}\right).$$

Hence we get

$$\mathbf{E}\left\{\sup_{\mathbf{w}\in\mathcal{W}}\left|\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\cdot(T_{\beta_n}(f_{\mathbf{w},1,1}^{(L_n)}(X_i))\right|\right\}$$

$$
\begin{aligned}
&\leq \delta_n + \int_{\delta_n}^{\infty} c_{12} \cdot \left(\frac{c_{13} \cdot \beta_n}{\delta_n} \right)^{c_{14} \cdot L_n^2 \cdot r^2 \cdot \log(L_n)} \cdot 2 \cdot \exp\left(- \frac{2 \cdot n \cdot t^2}{4 \cdot \beta_n^2} \right) dt \\
&\leq \delta_n + \int_{\delta_n}^{\infty} c_{12} \cdot \left(\frac{c_{13} \cdot \beta_n}{\delta_n} \right)^{c_{14} \cdot L_n^2 \cdot r^2 \cdot \log(L_n)} \cdot 2 \cdot \exp\left(- \frac{n \cdot \delta_n \cdot t}{2 \cdot \beta_n^2} \right) dt \\
&= \delta_n + c_{12} \cdot \left(\frac{c_{13} \cdot \beta_n}{\delta_n} \right)^{c_{14} \cdot L_n^2 \cdot r^2 \cdot \log(L_n)} \cdot \frac{4 \cdot \beta_n^2}{n \cdot \delta_n} \cdot \exp\left(- \frac{n \cdot \delta_n^2}{2 \cdot \beta_n^2} \right).
\end{aligned}
$$

With

$$\delta_n = L_n \cdot \sqrt{\log L_n} \cdot r \cdot \log n \cdot \sqrt{\frac{2 \cdot \beta_n^2}{n}}$$

we get the assertion. □

4.3 Approximation error

For $M \in \mathbb{N}$ let $\mathcal{P}$ be a partition of

$$\left[-\frac{1}{2M}, 1 + \frac{1}{2M} \right]^d$$

into $\bar{K} = (M+1)^d$ cubes $A_1, \ldots, A_{\bar{K}}$ with side length $1/M$. Denote the centers of these cubes by $a_1, \ldots, a_{\bar{K}}$ and set

$$\omega_k(x) = \prod_{j=1}^{d} \left(1 - M \cdot \left| a_k^{(j)} - x^{(j)} \right| \right)_+ \quad (x \in \mathbb{R}^d).$$

Then ω_k is a linear tensor product spline which is one at a_k and zero outside of

$$\left[a_k^{(1)} - \frac{1}{M}, a_k^{(1)} + \frac{1}{M} \right] \times \cdots \times \left[a_k^{(d)} - \frac{1}{M}, a_k^{(d)} + \frac{1}{M} \right]$$

(so-called hat function), and it is easy to see that we have

$$\sum_{k=1}^{\bar{K}} \omega_k(x) = 1 \quad (x \in [0,1]^d) \tag{21}$$

(cf., e.g., Lemma 15.2 in Györfi et al. (2002)). Let $p = q + s$ for some $q \in \mathbb{N}_0$ and $s \in (0,1]$, let $f : \mathbb{R}^d \to \mathbb{R}$ be a (p, C)-smooth function, let $x_0 \in \mathbb{R}^d$ and let

$$T_{f,q,x_0}(x) = \sum_{j \in \mathbb{N}_0^d, \|j\|_1 \leq q} (\partial^j f)(x_0) \cdot \frac{(x - x_0)^j}{j!}$$

be the Taylor polynomial of total degree q around x_0. Then we have for any $x \in \mathbb{R}^d$

$$|f(x) - T_{f,q,x_0}(x)| \leq c_{15} \cdot C \cdot \|x - x_0\|^p$$

(cf., e.g., Lemma 1 in Kohler (2014)), which implies

$$\begin{aligned}\left|f(x)-\sum_{k=1}^{\bar{K}}\omega_k(x)\cdot T_{f,q,a_k}(x)\right| &\leq \sum_{k=1}^{\bar{K}}\omega_k(x)\cdot|f(x)-T_{f,q,a_k}(x)| \\ &\leq \max_{k=1,\dots,\bar{K},\omega_k(x)\neq 0}|f(x)-T_{f,q,a_k}(x)| \\ &\leq c_{16}\cdot C\cdot\frac{1}{M^p}\quad (x\in[0,1]^d). \qquad (22)\end{aligned}$$

In this section we show that if we choose $w_{k,i,j}^{(l)}$ for $k=1,\dots,\binom{d+q}{d}\cdot(M+1)^d$ suitably and set $w_{1,1,k}^{(L_n)}=0$ for $k>\binom{d+q}{d}\cdot(M+1)^d$, then $f_{\mathbf{w}}$ approximates

$$x\mapsto\sum_{k=1}^{\bar{K}}\omega_k(x)\cdot T_{f,q,a_k}(x)$$

uniformly on $\mathbb{R}^d$, from which we conclude the following result.

Lemma 6 *Let $M\in\mathbb{N}$ with $M>1$, let $p=q+s$ for some $q\in\mathbb{N}_0$ and $s\in(0,1]$, let $C>0$ and let $f:\mathbb{R}^d\to\mathbb{R}$ be a (p,C)-smooth function. Set*

$$L=\lceil c_{17}\cdot\log M\rceil \qquad (23)$$

and

$$r=18\cdot(q+d)^{d+1}. \qquad (24)$$

Set

$$\tilde{K}=\binom{d+q}{d}\cdot(M+1)^d$$

and let $K_n\geq\tilde{K}$. Then there exists a weight vector $\mathbf{w}$ with

$$|w_{1,1,k}^{(L_n)}|\leq const(f)\quad (k=1,\dots,\tilde{K}), \qquad (25)$$

$$w_{k,i,0}^{(l)}=0\quad \textit{and}\quad w_{k,i,j}^{(l)}\in\left\{0,\frac{1}{2},1\right\}\quad (k=1,\dots,\tilde{K},l=1,\dots,L-1), \qquad (26)$$

$$\sum_{j=1}^{r}(w_{k,i,j}^{(l)})_+\leq 1\quad \textit{and}\quad \sum_{j=1}^{r}(w_{k,i,j}^{(l)})_-\leq 1\quad (l\in\{1,\dots,L-1\}), \qquad (27)$$

$$|w_{k,i,j}^{(0)}|\leq M+1\quad (k=1,\dots,\tilde{K}) \qquad (28)$$

and

$$w_{k,i,j}^{(l)}=0\quad \textit{for } k>\tilde{K}, \qquad (29)$$

such that for M, n sufficiently large it holds

$$\|f_{\mathbf{w}}-f\|_{\infty,[0,1]^d}\leq c_{18}\cdot C\cdot\frac{1}{M^p}.$$

Proof. Because of (22) it suffices to show that for a weight vector $\mathbf{w}$ satisfying (25)-(29) we have

$$\left| \sum_{k=1}^{(M+1)^d} \omega_k(x) \cdot T_{f,q,a_k}(x) - f_{\mathbf{w}}(x) \right| \leq c_{19} \cdot C \cdot \frac{1}{M^p} \quad (x \in [0,1]^d). \tag{30}$$

To show this, we show in the sequel that for any k and $j \in \mathbb{N}_0^d$ with $\|j\|_1 \leq q$ it is possible to choose $\mathbf{w}$ such that (25)-(28) and

$$\left| \omega_k(x) \cdot (x - a_k)^j - f_{\mathbf{w},1,1}^{(L_n)}(x) \right| \leq c_{20} \cdot C \cdot \frac{1}{M^{d+p}} \quad (x \in [0,1]^d) \tag{31}$$

hold. This implies the assertion, because if this is the case we can choose $\mathbf{w}$ such that (25)-(29) and

$$\begin{aligned} \left| \sum_{k=1}^{(M+1)^d} \omega_k(x) \cdot T_{f,q,a_k}(x) - \sum_{k=1}^{K_n} w_{1,1,k}^{(L_n)} \cdot f_{\mathbf{w},1,k}^{(L_n)}(x) \right| &\leq \sum_{k=1}^{\tilde{K}} c_{21} \cdot C \cdot \frac{1}{M^{d+p}} \\ &\leq c_{22} \cdot C \cdot \frac{1}{M^p}. \end{aligned}$$

Observe that (31) implies

$$|f_{\mathbf{w},1,k}^{(L_n)}(x)| \leq \beta_n$$

for M, n sufficiently large and hence it holds

$$f_{\mathbf{w}}(x) = \sum_{k=1}^{K_n} w_{1,1,k}^{(L_n)} \cdot f_{\mathbf{w},1,k}^{(L_n)}(x).$$

In order to show (31) we use in the following the neural network

$$\hat{f}_{mult,\binom{q+d}{d}+d} \in \mathcal{F}(R \cdot (d+1) \cdot \lceil \log_2(p+d) \rceil, 18 \cdot \lceil \log_2(p+d) \rceil)$$

from Lemma 8 in Kohler and Langer (2021) which satisfies

$$\left| \hat{f}_{mult,\binom{q+d}{d}+d}(z) - \prod_{i=1}^{\binom{p+d}{d}+d} z^{(i)} \right| \leq const(d,q) \cdot 4^{-R} \quad (z \in [0,1]^{\binom{q+d}{d}+d}).$$

By setting

$$R = const(d,q) \cdot \lceil \log M \rceil$$

and by applying this network to an argument z where the components $z^{(i)}$ $(i = 1, \ldots, \binom{q+d}{d})$ are chosen either from the values

$$x^{(j)} - a_k^{(j)} = \sigma(x^{(j)} - a_k^{(j)}) + \sigma(-x^{(j)} + a_k^{(j)}) \quad (j = 1, \ldots, d)$$

or are set equal to $1 = \sigma(1)$, and where

$$
\begin{aligned}
z^{(\binom{p+d}{d}+j)} &= \left(1 - M \cdot \left|a_k^{(j)} - x^{(j)}\right|\right)_+ \\
&= \sigma\left(M \cdot (x^{(j)} - a_k^{(j)} - \frac{1}{M})\right) - 2 \cdot \sigma\left(M \cdot (x^{(j)} - a_k^{(j)})\right) + \sigma\left(M \cdot (x^{(j)} - a_k^{(j)} + \frac{1}{M})\right)
\end{aligned}
$$

($j = 1, \dots, d$) we get the assertion. Here we have used that the weights in the network $\hat{f}_{mult,\binom{q+d}{d}+d}$ from Lemma 8 in Kohler and Langer (2021) can be chosen such that the conditions on the weights in Lemma 6 are satified. □

4.4 Proof of Theorem 1

Set

$$
M = M_n = \lceil c_{23} \cdot n^{\frac{1}{2\cdot(2p+d)}} \rceil
$$

and choose $L_n = L$ and r as in Lemma 6. Set

$$
\tilde{K}_n = \binom{d+q}{d} \cdot (M_n + 1)^d \approx c_{24} \cdot n^{\frac{d}{2\cdot(2p+d)}},
$$

$$
N_n = n^{c_{25}}
$$

and choose $w_{i,j,k}^{(l)}$ ($k \in \{1, \dots, \tilde{K}_n\}$) as in Lemma 6.

W.l.o.g. we assume throughout the proof that n is sufficiently large and that $\|m\|_\infty \leq \beta_n$ and $\beta_n \geq 1$ hold. Let A_n be the event that firstly the weight vector $\mathbf{w}^{(0)}$ satisfies

$$
|(\mathbf{w}^{(0)})^{(l)}_{j_{v,s},k,i} - \mathbf{w}^{(l)}_{s,k,i}| \leq \frac{1}{n^{c_{26}}} \quad \text{for all } l \in \{0, \dots, L-1\}, s \in \{1, \dots, \tilde{K}_n\}, v \in \{1, \dots, N_n\}
$$

for some pairwise distinct $j_{1,1}, \dots, j_{\tilde{K}_n, N_n} \in \{1, \dots, K_n\}$ and such that secondly

$$
\max_{i=1,\dots,n} |Y_i| \leq \beta_n
$$

holds.

Define the weight vectors $(\mathbf{w}^*)^{(t)}$ by

$$
((\mathbf{w}^*)^{(t)})^{(l)}_{k,i,j} = (\mathbf{w}^{(t)})^{(l)}_{k,i,j} \quad \text{for all } l = 0, \dots, L-1
$$

and

$$
((\mathbf{w}^*)^{(t)})^{(L_n)}_{1,1,j_{k,v}} = \frac{w_{1,1,k}^{(L_n)}}{N_n} \quad \text{for all } k = 1, \dots, \tilde{K}_n, \quad v = 1, \dots, N_n
$$

and

$$
((\mathbf{w}^*)^{(t)})^{(L_n)}_{1,1,k} = 0 \quad \text{for} \quad k \notin \{j_{\nu,s} : \nu = 1, \dots, \tilde{K}_n, \, s = 1, \dots, N_n\}.
$$

In the sequel we decompose the L_2 error of m_n in a sum of several terms. Set

$$
m_{\beta_n}(x) = \mathbf{E}\{T_{\beta_n} Y | X = x\}.
$$

We have

$$\begin{aligned}
&\int |m_n(x)-m(x)|^2 \mathbf{P}_X(dx) \\
&= \left(\mathbf{E}\left\{|m_n(X)-Y|^2|\mathcal{D}_n\right\} - \mathbf{E}\{|m(X)-Y|^2\}\right) \cdot 1_{A_n} \\
&\quad + \int |m_n(x)-m(x)|^2 \mathbf{P}_X(dx) \cdot 1_{A_n^c} \\
&= \Big[\mathbf{E}\left\{|m_n(X)-Y|^2|\mathcal{D}_n\right\} - \mathbf{E}\{|m_{\beta_n}(X)-T_{\beta_n}Y|^2\}\Big] \cdot 1_{A_n} \\
&\quad + \Big[\mathbf{E}\left\{|m_n(X)-T_{\beta_n}Y|^2|\mathcal{D}_n\right\} - \frac{1}{n}\sum_{i=1}^n |m_n(X_i)-T_{\beta_n}Y_i|^2\Big] \cdot 1_{A_n} \\
&\quad + \Big[\frac{1}{n}\sum_{i=1}^n |m_n(X_i)-T_{\beta_n}Y_i|^2 - \frac{1}{n}\sum_{i=1}^n |m_n(X_i)-Y_i|^2\Big] \cdot 1_{A_n} \\
&\quad + \Big[\frac{1}{n}\sum_{i=1}^n |m_n(X_i)-Y_i|^2 - \mathbf{E}\{|m(X)-Y|^2\}\Big] \cdot 1_{A_n} \\
&\quad + \int |m_n(x)-m(x)|^2 \mathbf{P}_X(dx) \cdot 1_{A_n^c} \\
&=: \sum_{j=1}^5 T_{j,n}.
\end{aligned}$$

In the reminder of the proof we bound

$$\mathbf{E}T_{j,n}$$

for $j \in \{1, \dots, 5\}$.

In the *first step of the proof* we show

$$\mathbf{E}T_{1,n} \leq c_{27} \cdot \frac{\log n}{n}.$$

This follows as in the proof of Lemma 1 in Bauer and Kohler (2019).

In the *second step of the proof* we observe

$$\mathbf{E}T_{3,n} \leq c_{28} \cdot \frac{\log n}{n}.$$

This holds trivially since we have on A_n

$$\frac{1}{n}\sum_{i=1}^n |m_n(X_i)-T_{\beta_n}Y_i|^2 = \frac{1}{n}\sum_{i=1}^n |m_n(X_i)-Y_i|^2.$$

In the *third step of the proof* we show

$$\mathbf{E}T_{5,n} \leq c_{29} \cdot \frac{(\log n)^2}{n}.$$

The definition of m_n implies $\int |m_n(x)-m(x)|^2 \mathbf{P}_X(dx) \leq 4 \cdot c_1^2 \cdot (\log n)^2$, hence it suffices to show

$$\mathbf{P}(A_n^c) \leq \frac{c_{30}}{n}. \tag{32}$$

To do this, we consider sequential choice of the weights of the K_n fully connected neural networks. Probability that the $(r+1)+(L_n-2)\cdot r\cdot(r+1)+r\cdot(d+1)$ many weights in the first of these networks differ in all components at most by $1/n^{c_{26}}$ from $w_{1,i,j}^{(l)}$ $(l=0,\dots,L-1)$ is for large n bounded below by

$$\left(\frac{1}{n^{c_{31}}}\right)^{(r+1)+(L_n-2)\cdot r\cdot(r+1)+r\cdot(d+1)} = \frac{1}{n^{c_{32}\cdot\log n}}.$$

Hence probability that none of the first $n^{c_{32}\cdot\log n+0.5}$ neural networks satisfies this condition is for large n bounded above by

$$\begin{aligned}\left(1-\frac{1}{n^{c_{32}\cdot\log n}}\right)^{n^{c_{32}\cdot\log n+0.5}} &\leq \left(\exp\left(-\frac{1}{n^{c_{32}\cdot\log n}}\right)\right)^{n^{c_{32}\cdot\log n+0.5}}\\ &= \exp(-n^{0.5}).\end{aligned}$$

Since we have $K_n \geq n^{c_{32}\cdot\log n+0.5}\cdot\tilde{K}_n\cdot N_n$ for n large we can successively use the same construction for all of $\tilde{K}_n\cdot N_n$ weights and we can conclude: Probability that there exists $k\in\{1,\dots,\tilde{K}_n\}$ such that not at least N_n of the K_n weight vectors of the fully connected neural network differs by at most $1/n^{c_{26}}$ from $(w_{i,j,k}^{(l)})_{i,j,l}$ is for large n bounded from above by

$$\tilde{K}_n\cdot N_n\cdot\exp(-n^{0.5}) \leq n^{c_{33}}\cdot\exp(-n^{0.5}) \leq \frac{c_{34}}{n}.$$

This implies for large n

$$\begin{aligned}\mathbf{P}(A_n^c) &\leq \frac{c_{34}}{n}+\mathbf{P}\{\max_{i=1,\dots,n}|Y_i|>\beta_n\} \leq \frac{c_{34}}{n}+n\cdot\mathbf{P}\{|Y|>\beta_n\}\\ &\leq \frac{c_{34}}{n}+n\cdot\frac{\mathbf{E}\{\exp(c_2\cdot Y^2)}{\exp(c_2\cdot\beta_n^2)} \leq \frac{c_{35}}{n},\end{aligned}$$

where the last inequality holds because of (9).

In the *fourth step of the proof* we show

$$\mathbf{E}T_{2,n} \leq c_{36}\cdot\frac{\gamma_n\cdot L_n\cdot\sqrt{\log L_n}\cdot\beta_n^2\cdot\log n}{\sqrt{n}}.$$

Define $\mathcal{F}$ as in Lemma 5. Then

$$\begin{aligned}\mathbf{E}T_{2,n} &= \mathbf{E}\left[\left[\mathbf{E}\left\{|m_n(X)-T_{\beta_n}Y|^2|\mathcal{D}_n\right\}-\frac{1}{n}\sum_{i=1}^n|m_n(X_i)-T_{\beta_n}Y_i|^2\right]\cdot 1_{A_n}\right]\\ &\leq \mathbf{E}\left[\left[\left(\mathbf{E}\left\{|m_n(X)-T_{\beta_n}Y|^2|\mathcal{D}_n\right\}-\frac{1}{n}\sum_{i=1}^n|m_n(X_i)-T_{\beta_n}Y_i|^2\right]\cdot 1_{A_n}\right)_+\right]\end{aligned}$$

$$
\begin{aligned}
&\leq \int_0^\infty \mathbf{P}\left\{ \mathbf{E}\left\{|m_n(X) - T_{\beta_n}Y|^2|\mathcal{D}_n\right\} - \frac{1}{n}\sum_{i=1}^n |m_n(X_i) - T_{\beta_n}Y_i|^2 > t \middle| \mathcal{D}_n \right\} dt \\
&\leq \mathbf{E}\left\{ \sup_{f \in \mathcal{F}} \left(\mathbf{E}\{|(T_{\beta_n}f)(X) - Y|^2\} - \frac{1}{n}\sum_{i=1}^n |(T_{\beta_n}f)(X_i) - Y_i|^2 \right) \right\},
\end{aligned}
$$

from which we get the assertion by an application of Lemma 5.

In the *fifth step of the proof* we show

$$
\mathbf{E}\{T_{4,n}\} \leq c_{37} \cdot \frac{1}{M_n^{2p}}.
$$

On A_n we have $|Y_i| \leq \beta_n$ $(i = 1, \dots, n)$. Because of $|T_{\beta_n}y - z| \leq |y - z|$ for $|z| \leq \beta_n$ this implies

$$
\begin{aligned}
\mathbf{E}\left\{ \frac{1}{n}\sum_{i=1}^n |m_n(X_i) - Y_i|^2 \cdot 1_{A_n} \right\} &\leq \mathbf{E}\left\{ \frac{1}{n}\sum_{i=1}^n |f_{\mathbf{w}^{(\hat{t})}}(X_i) - Y_i|^2 \cdot 1_{A_n} \right\} \\
&\leq \mathbf{E}\left\{ \min_{t=0,\dots,t_n} \frac{1}{n}\sum_{i=1}^n |f_{\mathbf{w}^{(t)}}(X_i) - Y_i|^2 \cdot 1_{A_n} \right\}.
\end{aligned}
$$

Furthermore by Lemma 4 we have on A_n

$$
\begin{aligned}
&\frac{1}{n}\sum_{i=1}^n |f_{(\mathbf{w}^*)^{(t)}}(X_i) - Y_i|^2 - \frac{1}{n}\sum_{i=1}^n |f_{(\mathbf{w}^*)^{(0)}}(X_i) - Y_i|^2 \\
&= \frac{1}{n}\sum_{i=1}^n \left(f_{(\mathbf{w}^*)^{(t)}}(X_i) - Y_i + f_{(\mathbf{w}^*)^{(0)}}(X_i) - Y_i \right) \cdot \left(f_{(\mathbf{w}^*)^{(t)}}(X_i) - f_{(\mathbf{w}^*)^{(0)}}(X_i) \right) \\
&\leq (2 \cdot \beta_n + 2 \cdot \gamma_n \cdot \beta_n) \cdot \sum_{k=1}^{K_n} |(\mathbf{w}^*)^{(t)}_{1,1,k}| \cdot |T_{\beta_n}(f_{(\mathbf{w}^*)^{(t)}_{k,1}}(X_i)) - T_{\beta_n}(f_{(\mathbf{w}^*)^{(0)}_{k,1}}(X_i))| \\
&\leq (2 \cdot \beta_n + 2 \cdot \gamma_n \cdot \beta_n) \cdot \sqrt{\sum_{k=1}^{K_n} |(\mathbf{w}^*)^{(t)}_{1,1,k}|^2} \\
&\qquad \cdot \sqrt{\sum_{k=1}^{K_n} |T_{\beta_n}(f_{(\mathbf{w}^*)^{(t)}_{k,1}}(X_i)) - T_{\beta_n}(f_{(\mathbf{w}^*)^{(0)}_{k,1}}(X_i))|^2 \cdot 1_{\{(\mathbf{w}^*)^{(t)}_{1,1,k} \neq 0\}}} \\
&\leq c_{38} \cdot (\log n) \cdot (4 \cdot r)^{L_n} \cdot (d+1) \cdot (M_n + 1) \cdot \left(1 + \frac{r+1}{n^{c_{20}}}\right)^{L_n} \cdot \sqrt{\sum_{k=1}^{K_n} |(\mathbf{w}^*)^{(t)}_{k,1,1}|^2} \\
&\qquad \cdot \|(\mathbf{w}^{(t)})_{i,j,k,l\,:\,l<L} - (\mathbf{w}^{(0)})_{i,j,k,l\,:\,l<L}\|.
\end{aligned}
$$

Together with Lemma 2 we can conclude that on A_n the assumptions of Lemma 1 are satisfied with

$$
C_n = c_{39} \cdot (\log n)^4 \cdot K_n \quad \text{and} \quad D_n = c_{40} \cdot (\log n) \cdot (4r)^{L_n} \cdot n^{\frac{1}{4 \cdot (2p+d)}}
$$

Application of Lemma 1 yields

$$\begin{aligned}
&\mathbf{E}\left\{\frac{1}{n}\sum_{i=1}^{n}|f_{\mathbf{w}^{(\hat{t})}}(X_i)-Y_i|^2\cdot 1_{A_n}\right\}\\
&\leq \mathbf{E}\Bigg\{\frac{1}{n}\sum_{i=1}^{n}|f_{(\mathbf{w}^*)^{(0)}}(X_i)-Y_i|^2+D_n\cdot\frac{1}{\sqrt{N_n}}\cdot\|(w_{k,1,1}^{(L)})_{k=1,\dots,\tilde{K}_n}\|\cdot 1\\
&\quad+\frac{1}{2}\cdot\frac{1}{N_n}\cdot\|(w_{k,1,1}^{(L)})_{k=1,\dots,\tilde{K}_n}\|^2+\frac{C_n^2}{2\cdot t_n}\Bigg\}.
\end{aligned}$$

Application of Lemma 6 yields

$$\begin{aligned}
&\mathbf{E}\{T_{4,n}\}=\mathbf{E}\{\mathbf{E}\{T_{4,n}|\mathbf{w}^{(0)}\}\}\\
&\leq \mathbf{E}\int|f_{(\mathbf{w}^*)^{(0)}}(x)-m(x)|^2\mathbf{P}_X(dx)+c_{37}\cdot\frac{\log n}{n}+4\cdot\beta_n^2\cdot\mathbf{P}(A_n)\\
&\leq 2\cdot\mathbf{E}\int|f_{(\mathbf{w}^*)^{(0)}}(x)-f_{(w_{i,j,k}^{(l)})_{i,j,k,l:k\leq\tilde{K}_n}}(x)|^2\mathbf{P}_X(dx)+c_{38}\cdot\frac{1}{M_n^{2p}}\\
&\leq c_{39}\cdot\frac{1}{M_n^{2p}},
\end{aligned}$$

where the last inequality followed by another application of Lemma 4. □

4.5 Proof of Theorem 2

Set

$$M=M_n=\lceil c_{40}\cdot n^{\frac{1}{2\cdot(2p+d^*)}}\rceil$$

and choose $L_n=L$ and r as in Lemma 6 with d replaced by d^*. Set

$$\tilde{K}_n=\binom{d}{d^*}\cdot\binom{d^*+q}{d^*}\cdot(M_n+1)^{d^*}\approx c_{41}\cdot n^{\frac{d^*}{2\cdot(2p+d^*)}},$$

$$N_n=n^{c_{42}}$$

and choose $w_{i,j,k}^{(l)}$ ($k\in\{1,\dots,\tilde{K}_n\}$) by using Lemma 6 in order to approximate

$$\sum_{I\subseteq\{1,\dots,d\}\,:\,|I|=d^*}m_I(x_I),$$

i.e., $f_{\mathbf{w}}$ is a sum of

$$\binom{d}{d^*}$$

many neural networks which approximate $m_I(x_I)$ ($I\subseteq\{1,\dots,d\}\,:\,|I|=d^*$). Define A_n as in the proof of Theorem 1 and use the same error decomposition as there. Then we get as in the proof of Theorem 1

$$\mathbf{E}T_{1,n}\leq c_{43}\cdot\frac{\log n}{n},$$

$$\mathbf{E}T_{2,n} \leq c_{44} \cdot \frac{\gamma_n \cdot L_n \cdot \sqrt{\log L_n} \cdot \beta_n \cdot \log n}{\sqrt{n}},$$

$$\mathbf{E}T_{3,n} \leq c_{45} \cdot \frac{\log n}{n}$$

and

$$\mathbf{E}T_{5,n} \leq c_{46} \cdot \frac{(\log n)^2}{n}.$$

Arguing as in the proof of Theorem 1 we see

$$\begin{aligned}&\mathbf{E}\{T_{4,n}\}\\ &\leq \int |f_{(\mathbf{w}^*)^{(0)}}(x) - m(x)|^2 \mathbf{P}_X(dx) + c_{47} \cdot \frac{\log n}{n} + 4 \cdot \beta_n^2 \cdot \mathbf{P}(A_n).\end{aligned}$$

Using that i.e., $f_{(\mathbf{w}^*)^{(0)}}(x)$ is a sum of

$$\binom{d}{d^*}$$

many neural networks which approximate $m_I(x_I)$ ($I \subseteq \{1, \dots, d\} : |I| = d^*$) according to Lemma 6 we get

$$|f_{(\mathbf{w}^*)^{(0)}}(x) - m(x)| \leq \sum_{I \subseteq \{1,\dots,d\} \,:\, |I|=d^*} |f_{(\mathbf{w}_I^*)^{(0)}}(x) - m(x_I)|| \leq \binom{d}{d^*} \cdot c_{48} \cdot \frac{1}{M_n^p}.$$

The assertion follows by summarizing the above results. □

References

[1] Andoni, A., Panigraphy, R., Valiant, G., and Zhang, L. (2014). Learning polynomials with neural networks. In *International Conference on Machine Learning*, pp. 1908–1916.

[2] Allen-Zhu, Z., Li, Y., und Song, Z. (2019). A convergence theory for deep learning via over-parameterization. *Proceedings of the 36th International Conference on Machine Learning (PMLR 2019)*, Long Beach, California, **97**, pp. 242-252.

[3] Barron, A. R. (1994). Approximation and estimation bounds for artificial neural networks. *Machine Learning* **14**, pp. 115-133.

[4] Bartlett, P., Harvey, N., Liaw, C., and Mehrabian, A. (2019). Nearly-tight VC-dimension bounds for piecewise linear neural networks. *Journal of Machine Learning Research* **20**, pp. 1-17.

[5] Bartlett, P. L., Montanari, A., and Rakhlin, A. (2021). Deep learning: a statistical viewpoint. Preprint, *arXiv: 2103.09177.*

[6] Bauer, B., and Kohler, M. (2019). On deep learning as a remedy for the curse of dimensionality in nonparametric regression. *Annals of Statistics* **4**, pp. 2261–2285.

[7] Birman, M. S., and Solomjak, M. Z. (1967). Piece-wise polynomial approximations of functions in the classes W_p^α. *Mathematics of the USSR Sbornik* **73**, pp. 295-317.

[8] Braun, A., Kohler, M., Langer, S., and Walk, H. (2023). Convergence rates for shallow neural networks learned by gradient descent. Accepted for publication in *Bernoulli.* Preprint, *arXiv: 2107.09550.*

[9] Chizat, L. and Bach, F. (2018). On the global convergence of gradient descent for over-parameterized models using optimal transport. Preprint, *arXiv: 1805.09545.*

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

[11] Du, S., Lee, J., Li, H., Wang, L., und Zhai, X. (2019). Gradient descent finds global minima of deep neural networks. International Conference on Machine Learning, Preprint, *arXiv: 1811.03804.*

[12] Györfi, L., Kohler, M., Krzyżak, A., and Walk, H. (2002). *A Distribution–Free Theory of Nonparametric Regression.* Springer.

[13] Golowich, N., Rakhlin, A., and Shamir, O. (2019). Size-Independent sample complexity of neural networks. Preprint, *arXiv: 1712.06541.*

[14] Gonon, L. (2021). Random feature networks learn Black-Scholes type PDEs without curse of dimensionality. Preprint, *arXiv: 2106.08900.*

[15] Hanin, B. and Nica, M. (2019). Finite depth and width corrections to the neural tangent kernel. Preprint, *arXiv: 1909.05989.*

[16] Huang, G. B., Chen, L., and Siew, C.-K. (2006). Universal approximation using incremental constructive feedforward networks with random hidden nodes. *IEEE Transactions on Neural Networks* **17**, pp. 879-892.

[17] Imaizumi, M., and Fukamizu, K. (2018). Deep neural networks learn non-smooth functions effectively. *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS 2019)*, Naha, Okinawa, Japan.

[18] Kawaguchi, K, and Huang, J. (2019). Gradient descent finds global minima for generalizable deep neural networks of practical sizes. *57th IEEE Annual Allerton Conference on Communication, Control, and Computing*, Allerton, IL, pp. 92-99.

[19] Kim, Y. (2014). Convolutional neural networks for sentence classification. Preprint, *arXiv: 1408.5882.*

[20] Kohler, M., and Krzyżak, A. (2017). Nonparametric regression based on hierarchical interaction models. *IEEE Transaction on Information Theory* **63**, pp. 1620-1630.

[21] Kohler, M., and Krzyżak, A. (2021). Over-parametrized deep neural networks minimizing the empirical risk do not generalize well. *Bernoulli* **27**, pp. 2564-2597. Preprint, *arXiv: 1912.03925.*

[22] Kohler, M., and Krzyżak, A. (2022). Analysis of the rate of convergence of an over-parametrized deep neural network estimate learned by gradient descent. Preprint, *arXiv: 2210.01443.*

[23] Kohler, M., and Langer, S. (2021). On the rate of convergence of fully connected deep neural network regression estimates using ReLU activation functions. *Annals of Statistics* **49**, pp. 2231-2249. Preprint, *arXiv: 1908.11133.*

[24] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In F. Pereira et al. (Eds.), *Advances In Neural Information Processing Systems* Red Hook, NY: Curran. **25**, pp. 1097-1105.

[25] Kutyoniok, G. (2020). Discussion of ”Nonparametric regression using deep neural networks with ReLU activation function”. *Annals of Statistics* **48**, pp. 1902–1905.

[26] Langer, S. (2021). Approximating smooth functions by deep neural networks with sigmoid activation function. *Journal of Multivariate Analysis* **182**, pp. 104696.

[27] Li, G., Gu, Y. and Ding, J. (2021). The Rate of Convergence of Variation-Constrained Deep Neural Networks. Preprint, *arXiv: 2106.12068*

[28] Liang, T., Rakhlin, A., and Sridharan, K. (2015). Learning with square loss: localization through offset Rademacher complexity. Preprint, *arXiv: 1502.06134.*

[29] Lin, S., and Zhang, J. (2019). Generalization bounds for convolutional neural networks. Preprint, *arXiv: 1910.01487.*

[30] Lu, J., Shen, Z., Yang, H., and Zhang, S. (2020). Deep network approximation for smooth functions. Preprint, *arXiv: 2001.03040*

[31] Mei, S., Montanari, A. and Nguyen, P.-M. (2018). A mean field view of the landscape of two-layer neural networks. In *Proceedings of the National Academy of Sciences*, **115**, pp. E7665-E7671.

[32] Nguyen, P.-M. and Pham, H. T. (2020). A Rigorous Framework for the Mean Field Limit of Multilayer Neural Networks Preprint, *arXiv: 2001.1144.*

[33] Rahimi, A., and Recht, B. (2008a). Random features for large-scale kernel machines. In *Advances in Neural Information Procesing Systems*, pp. 1177-1184.

[34] Rahimi, A., and Recht, B. (2008b). Uniform approximation of function with random bases. In *2008 46th Annual Allerton Conference on Communication, Control, and Computing*, pp. 555-561, IEEE.

[35] Rahimi, A., and Recht, B. (2009). Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In D. Koller, D. Schuurman, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, Curran Associates, Inc. **21**, pp. 1313-1320.

[36] Schmidt-Hieber, J. (2020). Nonparametric regression using deep neural networks with ReLU activation function (with discussion). *Annals of Statistics* **48**, pp. 1875–1897. Preprint, *arXiv:1708.06633v2.*

[37] Shamir, O., and Zhang, T. (2012). Stochastic gradient descent for non-smooth optimization: convergence results and optimal averaging schemes. Preprint, *arXiv: 1212.1824.*

[38] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Huber, T., et al. (2017). Mastering the game of go without human knowledge. *Nature* **550**, pp. 354-359.

[39] Stone, C. J. (1982). Optimal global rates of convergence for nonparametric regression. *Annals of Statistics*, **10**, pp. 1040-1053.

[40] Stone, C. J. (1994). The use of polynomial splines and their tensor products in multivariate function estimation. *Annals of Statistics*, **25**, pp. 118-184.

[41] Suzuki, T. (2018). Adaptivity of deep ReLU network for learning in Besov and mixed smooth Besov spaces: optimal rate and curse of dimensionality. Preprint, *arXiv: 1810.08033.*

[42] Suzuki, T., and Nitanda, A. (2019). Deep learning is adaptive to intrinsic dimensionality of model smoothness in anisotropic Besov space. Preprint, *arXiv: 1910.12799.*

[43] Wang, M., and Ma, C. (2022). Generalization error bounds for deep neural network trained by SGD. Preprint, *arXiv: 2206.03299v1.*

[44] Wu, Y., Schuster, M., Chen, Z., Le, Q., Norouzi, M., Macherey, W., Krikum, M., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. Preprint, *arXiv: 1609.08144.*

[45] Yarotsky, D. (2018). Optimal approximation of continuous functions by very deep ReLU networks. Preprint, *arXiv: 1802.03620*

[46] Yarotsky, D., and Zhevnerchuk, A. (2019). The phase diagram of approximation rates for deep neural networks. Preprint, *arXiv: 1906.09477.*

[47] Yehudai, G., and Shamir, O. (2022). On the power and limitations of random features for understanding neural networks. Preprint, *arXiv: 1904.00687*

[48] Zou, D., Cao, Y., Zhou, D., und Gu, Q. (2018). Stochastic gradient descent optimizes over-parameterized deep ReLU networks. Preprint, *arXiv: 1811.08888.*