Analysis of the rate of convergence of an over-parametrized convolutional neural network image classifier learned by gradient descent [*]

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

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

December 5, 2022

Abstract

Image classification based on over-parametrized convolutional neural networks with a global average-pooling layer is considered. The weights of the network are learned by gradient descent. A bound on the rate of convergence of the difference between the misclassification risk of the newly introduced convolutional neural network estimate and the minimal possible value is derived.

AMS classification: Primary 62G05; secondary 62G20.

Key words and phrases: convolutional neural networks, image classification, over-parametrization, rate of convergence.

1 Introduction

1.1 Scope of this paper

In deep learning, the task is to estimate the functional relationship between input and output using deep neural networks. For the particular application area of image classification, the input data consists of observed images and the output data represents classes of the corresponding images that describe what kind of objects are present in the images. The most successful methods, especially in the area of image classification can be attributed to deep learning approaches (see, e.g., Krizhevsky, Sutskever and Hinton (2012), LeCun, Bengio and Hinton (2015), and Rawat and Wang (2017)) and, in particular, to convolutional neural networks (CNNs). Recently, it has been shown that CNN image classifiers that minimize empirical risk are able to achieve dimension reduction (see Kohler, Krzyżak and Walter (2022), Kohler and Langer (2021), Walter (2021) and Kohler

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

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

and Walter (2022)). However, in practice, it is not possible to compute the empirical risk minimizer. Instead, gradient descent methods are used to obtain a small empirical risk. Furthermore, the network topologies used in practice are over-parameterized, i.e., they have many more trainable parameters than training samples.

The goal of this work is to derive the rate of convergence results for over-parameterized CNN image classifiers, which are trained by gradient descent. Thus this work should provide a better theoretical understanding of the empirical success of CNN image classifiers.

1.2 Image classification

We use the following statistical setting for image classification: Let $d_1, d_2 \in \mathbb{N}$ and let $(\mathbf{X}, Y)$, $(\mathbf{X}_1, Y_1)$, $\dots$, $(\mathbf{X}_n, Y_n)$ be independent and identically distributed random variables with values in

$$[0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}} \times \{0,1\}.$$

Here we use the notation

$$[0,1]^J = \{(a_j)_{j\in J} \, : \, a_j \in [0,1] \quad (j \in J)\}$$

for a nonempty and finite index set J, and we describe a (random) image from (random) class $Y \in \{0,1\}$ by a (random) matrix X with d_1 columns and d_2 rows, which contains at position (i,j) the grey scale value of the image pixel at the corresponding position.

Let

$$\eta(\mathbf{x}) = \mathbf{P}\{Y = 1 | \mathbf{X} = \mathbf{x}\} \quad (\mathbf{x} \in [0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}) \tag{1}$$

be the so–called a posteriori probability. Then we have

$$\min_{f:[0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}\to\{0,1\}} \mathbf{P}\{f(\mathbf{X}) \neq Y\} = \mathbf{P}\{f^*(\mathbf{X}) \neq Y\},$$

where

$$f^*(\mathbf{x}) = \begin{cases} 1, & \text{if } \eta(\mathbf{x}) > \frac{1}{2} \\ 0, & \text{elsewhere} \end{cases}$$

is the so–called Bayes classifier (cf., e.g., Theorem 2.1 in Devroye, Györfi and Lugosi (1996)). Set

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

In the sequel we consider the problem of constructing a classifier

$$f_n = f_n(\cdot, \mathcal{D}_n) : [0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}} \to \{0,1\}$$

such that the misclassification risk

$$\mathbf{P}\{f_n(\mathbf{X}) \neq Y | \mathcal{D}_n\}$$

of this classifier is as small as possible. Our aim is to derive a bound on the expected difference of the misclassification risk of f_n and the optimal misclassification risk, i.e., we want to derive an upper bound on

$$\begin{aligned}&\mathbf{E}\left\{\mathbf{P}\{f_n(\mathbf{X}) \neq Y|\mathcal{D}_n\} - \min_{f:[0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}\to\{0,1\}} \mathbf{P}\{f(\mathbf{X}) \neq Y\}\right\}\\ &= \mathbf{P}\{f_n(\mathbf{X}) \neq Y\} - \mathbf{P}\{f^*(\mathbf{X}) \neq Y\}.\end{aligned}$$

It is well-known that one needs to impose regularity conditions on the underlying distribution in order to derive non-trivial rate of convergence results for the error of the misclassification risk of any estimate in pattern recognition (cf., e.g., Cover (1968) and Devroye and Wagner (1980)). In the sequel we will assume that our a posteriori probability satisfies the model introduced below (see Definition 1), which is a modification of the generalized hierarchical max-pooling model introduced in Kohler, Krzyżak and Walter (2022).

The generalized hierarchical max-pooling model, which is also used in slightly modified form in Kohler and Langer (2021), Walter (2021), and in Kohler and Walter (2022), is motivated by the two ideas that, firstly, the object to be classified is contained in a subpart of the image and, secondly, that an image is hierarchically composed of neighboring subparts. The first idea is realized by looking at each subpart of the image and estimating for each subpart the probability that it contains the corresponding object. It is then assumed that the probability for the entire image corresponds to the maximum of the probabilities of all subparts of the image. The difference between the previous model and our new model is that instead of the maximum, we compute an average over all subparts (see Definition 1 a)). The advantage here is that our new model includes classification tasks where an image must contain multiple objects at possibly different image positions, but also classification tasks where only a single object has to be detected (in case that at each subpart the product of the probability that the subpart contains the object and a constant greater than 1 is estimated). The second idea realized in the hierarchical max-pooling model is that the probability for a subpart of the image is hierarchically composed of decisions of smaller neighboring subparts. This idea is not realized in our new model introduced below.

Definition 1 Let $d_1, d_2 \in \mathbb{N}$ with $d_1, d_2 > 1$ and $m : [0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}} \to \mathbb{R}$.
a) Let $\kappa \in \mathbb{N}$ with $\kappa \leq \min\{d_1, d_2\}$ and set

$$I = \{0, \dots, \kappa - 1\} \times \{0, \dots, \kappa - 1\}.$$

We say that m satisfies a **average-pooling model with size** κ^2, if there exists a function $f : [0,1]^{(1,1)+I} \to \mathbb{R}$ such that

$$m(\mathbf{x}) = \frac{1}{(d_1 - \kappa + 1) \cdot (d_2 - \kappa + 1)} \cdot \sum_{(i,j)\in\mathbb{Z}^2\,:\,(i,j)+I\subseteq\{1,\dots,d_1\}\times\{1,\dots,d_2\}} f\left(\mathbf{x}_{(i,j)+I}\right).$$

b) Let $p \in (0, \infty)$. We say that a **average-pooling model of size** κ^2 **has smoothness constraint** p, if the function f in the definition of m is (p, C)–smooth for some $C > 0$ (see Subsection 1.6 for the definition of (p, C)–smoothness).

1.3 Convolutional neural networks

The starting point in the construction of our estimate are convolutional neural networks with $L \in \mathbb{N}$ convolutional layers, one linear layer and one average-pooling layer for a $[0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}$–valued input, where $d_1, d_2 \in \mathbb{N}$. These networks have $k_r \in \mathbb{N}$ channels (also called feature maps) in the convolutional layer r and the convolution in layer r is performed by a window of values of layer $r-1$ of size $M_r \in \{1,\dots,\min\{d_1,d_2\}\}$, where $r \in \{1,\dots,L\}$. We will denote the input layer as the convolutional layer 0 with $k_0 = 1$ channels. The average-pooling layer will depend on a parameter $M_{L+1} \in \{1,\dots,\min\{d_1,d_2\}\}$ which describes the size of the window over which the output of layer L is averaged.

The networks depend on the weight matrix (so–called filter)

$$\mathbf{w} = \left(w^{(r)}_{i,j,s_1,s_2}\right)_{1\leq i,j\leq M_r, s_1\in\{1,\dots,k_{r-1}\}, s_2\in\{1,\dots,k_r\}, r\in\{1,\dots,L\}},$$

the weights

$$\mathbf{w}_{bias} = \left(w^{(r)}_{s_2}\right)_{s_2\in\{1,\dots,k_r\}, r\in\{1,\dots,L\}}$$

for the bias in each channel and each convolutional layer and the output weights

$$\mathbf{w}_{out} = (w_s)_{s\in\{1,\dots,k_L\}}.$$

For given weight vectors $\mathbf{w}$, $\mathbf{w}_{bias}$ and $\mathbf{w}_{out}$ the output of the networks is given by a real–valued function on $[0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}$ of the form

$$\begin{aligned} & f_{\mathbf{w},\mathbf{w}_{bias},\mathbf{w}_{out}}(\mathbf{x}) \\ & = \frac{1}{(d_1 - M_{L+1} + 1)\cdot(d_2 - M_{L+1} + 1)} \cdot \sum_{\substack{i\in\{1,\dots,d_1-M_{L+1}+1\},\\ j\in\{1,\dots,d_2-M_{L+1}+1\}}} \left(\sum_{s_2=1}^{k_L} w_{s_2}\cdot o^{(L)}_{(i,j),s_2}\right), \end{aligned}$$

where $o^{(L)}_{(i,j),s_2}$ is the output of the last convolutional layer, which is recursively defined as follows:

We start with

$$o^{(0)}_{(i,j),1} = x_{i,j} \quad \text{for } i \in \{1,\dots,d_1\} \text{ and } j \in \{1,\dots,d_2\}.$$

Then we define recursively

$$o^{(r)}_{(i,j),s_2} = \sigma\left(\sum_{s_1=1}^{k_{r-1}} \sum_{\substack{t_1,t_2\in\{1,\dots,M_r\}\\ (i+t_1-1,j+t_2-1)\in D}} w^{(r)}_{t_1,t_2,s_1,s_2}\cdot o^{(r-1)}_{(i+t_1-1,j+t_2-1),s_1} + w^{(r)}_{s_2}\right) \tag{2}$$

for the index set $D = \{1,\dots,d_1\}\times\{1,\dots,d_2\}$, $(i,j)\in D$, $s_2 \in \{1,\dots,k_r\}$ and $r \in \{1,\dots,L\}$, where $\sigma : \mathbb{R}\to\mathbb{R}$ is the activation function of the convolutional neural network, for which we will use throughout this paper the logistic squasher defined by $\sigma(x) = 1/(1+e^{-x})$.

In this paper we consider a special topology of the network where we compute a huge number of the above convolutional networks in parallel and the output of the network is then defined as a linear combination of the outputs of all those networks. Here all weights (including the weights used in the linear combination of the networks) will be learned by gradient descent starting with some proper (random) initialization, cf. Section 2 concerning the details.

1.4 Main result

In this paper we introduce an over-parametrized convolutional neural network image classifier where all weights are learned by gradient descent. We show that in case that the a posteriori probability satisfies conditions of an average-pooling model of size κ^2 and with smoothness constraint $p \in [1/2, 1]$, a proper random initialization of our weights together with proper choices of the stepsize and the number of gradient descent steps results for any $\epsilon > 0$ in

$$\begin{aligned} &\mathbf{P}\{f_n(\mathbf{X}) \neq Y\} - \min_{f:[0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}} \to \{0,1\}} \mathbf{P}\{f(\mathbf{X}) \neq Y\} \\ &\leq c_1 \cdot n^{-\frac{1}{2\cdot\kappa^2+2}+\epsilon}. \end{aligned}$$

The upper bound above on the the difference between the misclassification risk of the newly introduced convolutional neural network estimate and the minimal possible risk (Bayes risk) does not depend on the dimension (d_1, d_2) of the image, which shows that our convolutional neural network estimate is able to achieve some kind of dimension reduction. As far as we know the above result is the first rate of convergence result derived for convolutional neural network estimates where the weights are learned by gradient descent (using only one single random initialization). Our proof relies on the techniques recently developed by Drews and Kohler (2022) and Kohler and Krzyżak (2022) for the analysis of over-parametrized deep feedforward neural networks learned by gradient descent and our main achievement is to demonstrate that these techniques can also be used to analyze the rates of convergence of over-parametrized convolutional neural network estimates learned by gradient descent.

1.5 Discussion of related results

Deep neural networks have been studied intensively in the last decade and applied widely in different domains, see Goodfellow, Bengio and Courville (2016). Theoretical analysis of deep network learning has been actively pursued in recent years, see Berner et al. (2021) for a recent survey of progress in mathematics of deep learning. Among different deep network architectures convolutional neural networks introduced by LeCun (1989) are the most popular. They have been applied in image classification by Krizhevsky, Sutskever and Hinton (2012) and Kohler, Krzyżak and Walter (2022). In the latter paper the authors investigated the rates of CNN image classifiers.

Several recent papers demonstrated theoretically that backpropagation learning works for deep neural networks. The most popular approach which emerged in this context is so–called landscape approach. Choromanska et al. (2015) used random matrix theory to derive a heuristic argument showing that the risk of most of the local minima of the empirical L_2 risk $F_n(\mathbf{w})$ is not much larger than the risk of the global minimum. This claim was validated for neural networks with special activation function by, e.g., Arora et al. (2018), Kawaguchi (2016), and Du and Lee (2018), which have analyzed gradient descent for neural networks with a linear or quadratic activation function. No good approximation results exist for such neural networks, and consequently one cannot deduce from these results good rates of convergence for neural network regression estimates. Du et al. (2018) analyzed gradient descent learning for neural networks with one hidden layer and Gaussian inputs. As they used the expected gradient instead of the gradient in their gradient descent routine, one cannot apply their results to derive the rate of convergence for neural network regression estimates learned by the gradient descent. Liang et al. (2018) applied gradient descent to a modified loss function in classification, where it is assumed that the data can be interpolated by a neural network. Neural tangent kernel networks (NTK) were introduced by Jacot, Gabriel and Honger (2020). They showed that in the infinite-width limit case NTK converges to a deterministic limit kernel which stays constant during Gaussian descent training of the random weights initialized with the Gaussian distributions. These results were extended by Huang, Du and Xu (2020) to orthogonal initialization which was shown to speed up training of fully connected deep networks. Nitanda and Suzuki (2017) obtained global convergence rate for the averaged stochastic gradient descent for over-parametrized shallow neural networks. Braun et al. (2021) showed rate of convergence $1/\sqrt{n}$ (up to a logarithmic factor) for regression functions that have Fourier transforms with polynomially decreasing tails (an assumption slightly stronger than the finite first moment of the Fourier transform assumption of Barron (1993)).

Recently it was shown in several papers, see, e.g., Allen-Zhu, Li and Song (2019), Kawaguchi and Huang (2019) and the literature cited therein, that the gradient descent leads to a small empirical L_2 risk in over-parametrized neural networks. Here the results in Allen-Zhu, Li and Song (2019) are proven for the ReLU activation function and neural networks with a polynomial size in the sample size. The neural networks in Kawaguchi and Huang (2019) use squashing activation functions and are much smaller (in fact, they require only a linear size in the sample size). In contrast to Allen-Zhu, Li and Song (2019) there the learning rate is set to zero for all neurons except for neurons in the output layer and consequently in different layers of the network different learning rates are used. Actually, they compute a linear least squares estimate with the gradient descent, which is not used in practice. It was shown in Kohler and Krzyżak (2021) that any estimate which interpolates the training data does not generalize well in a sense that it can, in general, not achieve the optimal minimax rate of convergence in case of a general design measure. In recent survey paper Bartlett, Montanari and Rakhlin (2021) conjectured that over-parametrization allows gradient descent to find interpolating solutions which implicitly impose regularization, and that over-parametrization leads to benign overfitting. For related results involving the truncated Hilbert kernel regression estimate refer to Belkin, Rakhlin and Tsybakov (2019) and to Wyner et al. (2017) for the results involving AdaBoost and random forests. Linear regression in overfitting regime has been also considered in Bartlett, Long and Lugosi (2020). Benign overparametrization in shallow ReLU networks has been analyzed by Wang and Lin (2021). They showed L_2 error rate of $\sqrt{\log n/n}$ for over-parametrized neural network when the number of hidden neurons exceeds the number of samples.

Overparametrized deep neural network multivariable regression function estimates have been analyzed in recent papers. Universal consistency of such estimates was shown for over-parameterized standard deep feedforward neural networks learned by gradient descent by Drews and Kohler (2022). This paper was generalized by Kohler and Krzyżak (2022), who studied the rates of convergence. The approach used in the present paper is related to these two papers. In our proof we control the complexity of our over-parametrized convolutional neural networks by using metric entropy bounds as in Li, Gu and Ding (2021). A different approach based on the Rademacher complexity is presented in Wang and Ma (2022).

1.6 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 denote the greatest integer less than or equal to z by $\lfloor z \rfloor$. The Euclidean norm of $x \in \mathbb{R}^d$ is denoted by $\|x\|$, and we set

$$\|x\|_\infty = \max\{|x^{(1)}|, \dots, |x^{(d)}|\}$$

for $x = (x^{(1)}, \dots, x^{(d)})^T \in \mathbb{R}^d$. For $f : \mathbb{R}^d \to \mathbb{R}$

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

is its supremum norm. A function $f : \mathbb{R}^d \to \mathbb{R}$ is called (p, C)-smooth, if for every $\alpha = (\alpha_1, \dots, \alpha_d) \in \mathbb{N}_0^d$ with $\sum_{j=1}^d \alpha_j = q$ the partial derivative $\frac{\partial^q f}{\partial x_1^{\alpha_1} \dots \partial x_d^{\alpha_d}}$ exists and satisfies

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

for all $\mathbf{x}, \mathbf{z} \in \mathbb{R}^d$.

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–cover of $\mathcal{F}$ on x_1^n if for any $f \in \mathcal{F}$ there exists $i \in \{1, \dots, N\}$ such that

$$\left(\frac{1}{n} \sum_{k=1}^n |f(x_k) - f_i(x_k)|^p \right)^{1/p} < \varepsilon.$$

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

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 and $\mathcal{F}$ is a set of such functions, then we set $(T_\beta f)(x) = T_\beta(f(x))$.

1.7 Outline

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

2 Definition of the estimate

Throughout the paper we let $\sigma(x) = 1/(1+e^{-x})$ be the logistic squasher and we define the topology of our convolutional neural networks as follows: We compute a large number $K_n \in \mathbb{N}$ of the convolutional neural networks in Subsection 1.3 in parallel, where for simplicity we use for each of these networks $k_L = 1$ and $w_1 = 1$ (i.e., we skip the linear combination before the average-pooling), and we compute a linear combination of the output of these K_n convolutional neural networks. Here we use again $k_0 = 1$.

We set

$$f_{\mathbf{w}}(\mathbf{x}) = \sum_{k=1}^{K_n} w_k \cdot f_{\mathbf{w}_k, \mathbf{w}_{bias,k}}(\mathbf{x}) \tag{3}$$

where for $k \in \{1, \dots, K_n\}$

$$f_{\mathbf{w}_k, \mathbf{w}_{bias,k}}(\mathbf{x}) = \frac{1}{(d_1 - M_{L+1} + 1) \cdot (d_2 - M_{L+1} + 1)} \cdot \sum_{\substack{i \in \{1, \dots, d_1 - M_{L+1} + 1\}, \\ j \in \{1, \dots, d_2 - M_{L+1} + 1\}}} o_{(i,j),1,k}^{(L)}, \tag{4}$$

$$o_{(i,j),s_2,k}^{(r)} = \sigma\left(\sum_{s_1=1}^{k_{r-1}} \sum_{\substack{t_1, t_2 \in \{1, \dots, M_r\} \\ (i+t_1-1, j+t_2-1) \in D}} w_{t_1,t_2,s_1,s_2,k}^{(r)} \cdot o_{(i+t_1-1, j+t_2-1),s_1,k}^{(r-1)} + w_{s_2,k}^{(r)} \right) \tag{5}$$

$((i,j) \in D = \{1, \dots, d_1\} \times \{1, \dots, d_2\}, s_2 \in \{1, \dots, k_r\}, r \in \{1, \dots, L\})$ and

$$o_{(i,j),1,k}^{(0)} = x_{i,j} \quad \text{for } (i,j) \in D. \tag{6}$$

Let $\mathbf{w}$ be the vector of all the weights of the above network, i.e., $\mathbf{w}$ contains $w_1, \dots, w_{K_n}$ together with all weights $w_{t_1,t_2,s_1,s_2,k}^{(r)}$, $w_{s_2,k}^{(r)}$. We want to choose $\mathbf{w}$ such that the misclassification risk of $f_{\mathbf{w}}$ is small. To achieve this, we first estimate the a posteriori probability η by a network $f_{\mathbf{w}}$ which has a small empirical L_2 risk

$$\frac{1}{n} \sum_{i=1}^{n} |Y_i - f_{\mathbf{w}}(\mathbf{X}_i)|^2 \tag{7}$$

and then use the corresponding plug-in classificator for our image classification problem.

Minimization of (7) with respect to $\mathbf{w}$ is a nonlinear minimization problem which, in general, cannot be solved exactly. In the sequel we use gradient descent to obtain an approximate solution to the minimization problem.

We start with a random initialization of $\mathbf{w}$. We define $\mathbf{w}^{(0)}$ by setting

$$(\mathbf{w}^{(0)})_k = 0 \quad (k = 1, \dots, K_n)$$

and by choosing all other components of $\mathbf{w}^{(0)}$ as independent random variables sampled from some uniform distributions. Here $w_{t_1,t_2,s_1,s_2,k}^{(r)}$ and $w_{s_2,k}^{(r)}$ are uniformly distributed on $[-c_2 \cdot (\log n)^2, c_2 \cdot (\log n)^2]$ for $r = 2, \dots, L$, and in case $r = 1$ they are uniformly distributed on

$$[-c_3 \cdot (\log n)^2 \cdot n^\tau, c_3 \cdot (\log n)^2 \cdot n^\tau],$$

where $\tau > 0$ is a parameter of the estimate, which will be chosen in Theorem 1 below.

Then we use gradient descent to define recursively weight vectors $\mathbf{w}^{(t)}$ for $t = 1, \dots, t_n$. Here we add a regularization term to the empirical L_2 risk (7), i.e., we define

$$F_n(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^{n} |Y_i - f_{\mathbf{w}}(\mathbf{X}_i)|^2 + c_4 \cdot \sum_{k=1}^{K_n} w_k^2, \tag{8}$$

and apply gradient descent in order to minimize $F_n(\mathbf{w})$ with respect to $\mathbf{w}$, i.e., we compute

$$\mathbf{w}^{(t)} = \mathbf{w}^{(t-1)} - \lambda_n \cdot \nabla_{\mathbf{w}} F_n(\mathbf{w}^{(t-1)}) \quad (t = 1, \dots, t_n). \tag{9}$$

Here $\lambda_n > 0$ is the stepsize and $t_n \in \mathbb{N}$ is the number of gradient descent steps, and both will be chosen in Theorem 1 below.

Finally we define our image classifier f_n as the plug-in classifier corresponding to $f_{\mathbf{w}^{(t_n)}}$, i.e., we set

$$f_n(\mathbf{x}) = \begin{cases} 1, & \text{if } f_{\mathbf{w}^{(t_n)}}(\mathbf{x}) \geq \frac{1}{2}, \\ 0, & \text{elsewhere.} \end{cases} \tag{10}$$

3 Main result

Our main result is the following bound on the difference between the misclassification risk of our estimator and the optimal misclassification risk.

Theorem 1 *Let $d_1, d_2, \kappa \in \mathbb{N}$ with $\kappa \leq \min\{d_1, d_2\}$. Let $(\mathbf{X}, Y)$, $(\mathbf{X}_1, Y_1)$, ..., $(\mathbf{X}_n, Y_n)$ be independent and identically distributed $[0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}} \times \{0,1\}$-valued random variables. Assume that the a posteriori probability $\eta(\mathbf{x}) = \mathbf{P}\{Y = 1|\mathbf{X} = \mathbf{x}\}$ satisfies a average-pooling model of size κ^2 with smoothness constraint $p \in \left[\frac{1}{2}, 1\right]$. Choose*

$$L \geq 2$$

and $K_n \in \mathbb{N}$ such that

$$\frac{K_n}{n^{2\cdot\kappa^2+7}} \to \infty \quad (n \to \infty) \tag{11}$$

and

$$\frac{K_n}{n^\rho} \to 0 \quad (n \to \infty) \tag{12}$$

for some $\rho > 0$ hold. Choose $L_n \in \mathbb{N}$ with

$$L_n \geq (\log n)^{6L+2} \cdot K_n^{3/2},$$

set

$$\lambda_n = \frac{1}{L_n} \quad \textit{and} \quad t_n = \lceil c_5 \cdot (\log n) \cdot L_n \rceil,$$

$$\tau = \frac{1}{1+\kappa^2}$$

and

$$M_1 = M_{L+1} = \kappa, \quad M_2 = \cdots = M_L = 1, \quad k_1 = \cdots = k_{L-1} = 2 \cdot \kappa^2 \quad \textit{and} \quad k_0 = k_L = 1. \tag{13}$$

Assume

$$c_5 \geq \frac{1}{2 \cdot c_4}.$$

Define the estimate as in Section 2. Then we have for any $\epsilon > 0$

$$\mathbf{P}\{f_n(\mathbf{X}) \neq Y\} - \min_{f:[0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}} \to \{0,1\}} \mathbf{P}\{f(\mathbf{X}) \neq Y\}$$
$$\leq c_6 \cdot n^{-\frac{1}{2\cdot\kappa^2+2}+\epsilon}$$

for some constant $c_6 > 0$ which does not depend on d_1, d_2 and n.

Remark 1. The above rate of convergence does not depend on the dimension (d_1, d_2) of the image, instead it depends only on the parameter κ^2 (where $\kappa \leq \min\{d_1, d_2\}$) of the average–pooling model for η. Hence in case that the a posteriori probability η satisfies an average–pooling model, our convolutional neural network estimate is able to circumvent the curse of dimensionality.

Remark 2. In the proof of Theorem 1 we show that a truncated version $\hat{\eta}_n$ of the convolutional neural network $f_{\mathbf{w}^{(t_n)}}$ satisfies

$$\mathbf{E} \int |\hat{\eta}_n(\mathbf{x}) - \eta(\mathbf{x})|^2 \mathbf{P}_X(d\mathbf{x}) \leq c_7 \cdot n^{-\frac{1}{\kappa^2+1}+\epsilon}.$$

According to Stone (1982), the optimal minimax rate of convergence for estimation of a d-dimensional (p, C)–smooth regression function is

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

Hence our truncated version $\hat{\eta}_n$ of the convolutional neural network $f_{\mathbf{w}^{(t_n)}}$ achieves a rate of convergence which is close to the optimal minimax rate of convergence for estimation of a κ^2-dimensional $(1/2, C)$–smooth regression function.

4 Proofs

4.1 Auxiliary results

Lemma 1 *Define* $(\mathbf{X}, Y)$, $(\mathbf{X}_1, Y_1)$, ..., $(\mathbf{X}_n, Y_n)$, $\mathcal{D}_n$, η, *and* f^* *as in Subsection 1.2. Let*

$$\eta_n(\cdot) = \eta_n(\cdot, \mathcal{D}_n) : [0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}} \to \mathbb{R}$$

be an estimate of η *and set*

$$f_n(\mathbf{x}) = \begin{cases} 1, & \textit{if } \eta_n(\mathbf{x}) \geq \frac{1}{2}, \\ 0, & \textit{elsewhere.} \end{cases}$$

Then

$$\begin{aligned} \mathbf{P}\{f_n(\mathbf{X}) \neq Y | \mathcal{D}_n\} - \mathbf{P}\{f^*(\mathbf{X}) \neq Y\} &\leq 2 \cdot \int |\eta_n(\mathbf{x}) - \eta(\mathbf{x})| \, \mathbf{P}_{\mathbf{X}}(d\mathbf{x}) \\ &\leq 2 \cdot \sqrt{\int |\eta_n(x) - \eta(x)|^2 \mathbf{P}_{\mathbf{X}}(dx)} \end{aligned}$$

holds.

Proof. See Theorem 1.1 in Györfi et al. (2002). □

Lemma 2 *Let* $F : \mathbb{R}^K \to \mathbb{R}_+$ *be a nonnegative differentiable function. Let* $t \in \mathbb{N}$, $L > 0$, $\mathbf{a}_0 \in \mathbb{R}^K$ *and set*

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

and

$$\mathbf{a}_{k+1} = \mathbf{a}_k - \lambda \cdot (\nabla_{\mathbf{a}} F)(\mathbf{a}_k) \quad (k \in \{0, 1, \dots, t-1\}).$$

Assume

$$\|(\nabla_{\mathbf{a}} F)(\mathbf{a})\| \leq \sqrt{2 \cdot t \cdot L \cdot \max\{F(\mathbf{a}_0), 1\}} \tag{14}$$

for all $\mathbf{a} \in \mathbb{R}^K$ *with* $\|\mathbf{a} - \mathbf{a}_0\| \leq \sqrt{2 \cdot t \cdot \max\{F(\mathbf{a}_0), 1\}/L}$, *and*

$$\|(\nabla_{\mathbf{a}} F)(\mathbf{a}) - (\nabla_{\mathbf{a}} F)(\mathbf{b})\| \leq L \cdot \|\mathbf{a} - \mathbf{b}\| \tag{15}$$

for all $\mathbf{a}, \mathbf{b} \in \mathbb{R}^K$ *satisfying*

$$\|\mathbf{a} - \mathbf{a}_0\| \leq \sqrt{8 \cdot \frac{t}{L} \cdot \max\{F(\mathbf{a}_0), 1\}} \quad \textit{and} \quad \|\mathbf{b} - \mathbf{a}_0\| \leq \sqrt{8 \cdot \frac{t}{L} \cdot \max\{F(\mathbf{a}_0), 1\}}. \tag{16}$$

Then we have

$$\|\mathbf{a}_k - \mathbf{a}_0\| \leq \sqrt{2 \cdot \frac{k}{L} \cdot (F(\mathbf{a}_0) - F(\mathbf{a}_k))} \quad \textit{for all } k \in \{1, \dots, t\},$$

$$\sum_{k=0}^{s-1} \|\mathbf{a}_{k+1} - \mathbf{a}_k\|^2 \leq \frac{2}{L} \cdot (F(\mathbf{a}_0) - F(\mathbf{a}_s)) \quad \textit{for all } s \in \{1, \dots, t\}$$

and

$$F(\mathbf{a}_k) \leq F(\mathbf{a}_{k-1}) - \frac{1}{2L} \cdot \|\nabla_{\mathbf{a}} F(\mathbf{a}_{k-1})\|^2 \quad \textit{for all } k \in \{1, \dots, t\}.$$

Proof. The result follows from Lemma 2 in Braun et al. (2021) and its proof. □

Lemma 3 *Let $\sigma : \mathbb{R} \to \mathbb{R}$ be bounded and differentiable, and assume that its derivative is bounded. Let $t_n \geq L_n$, $\gamma_n^* \geq 1$, $B_n \geq 1$,*

$$|w_k| \leq \gamma_n^* \quad (k = 1, \dots, K_n), \tag{17}$$

$$|w_{s_2,k}^{(r)}| \leq B_n \text{ and } |w_{t_1,t_2,s_1,s_2,k}^{(r)}| \leq B_n \quad \text{for } r = 2, \dots, L \tag{18}$$

and

$$\|\mathbf{w} - \mathbf{v}\|_\infty^2 \leq \frac{2t_n}{L_n} \cdot \max\{F_n(\mathbf{v}), 1\}. \tag{19}$$

Assume $X_1, \dots, X_n \in [0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}$ and define F_n by (8), where parameters $L \in \mathbb{N}$, $M_1, \dots, M_{L+1} \in \mathbb{N}$ and $k_0, \dots, k_L \in \mathbb{N}$ of the convolutional neural network used in (8) satisfy $L \geq 2$, $M_2 = \dots = M_L = 1$, $M_1 = M_{L+1} = \kappa$, $k_1 = \dots = k_{L-1} = 2 \cdot \kappa^2$ and $k_0 = k_L = 1$.

Then we have

$$\|(\nabla_{\mathbf{w}} F_n)(\mathbf{w})\| \leq c_8 \cdot K_n^{3/2} \cdot B_n^{2L} \cdot (\gamma_n^*)^2 \cdot \sqrt{\frac{t_n}{L_n} \cdot \max\{F_n(\mathbf{v}), 1\}}.$$

Proof. Because of $M_2 = \dots = M_L = 1$, $M_1 = M_{L+1} = \kappa$, $k_1 = \dots = k_{L-1} = 2 \cdot \kappa^2$ and $k_0 = 1$ we have

$$f_{\mathbf{w}}(\mathbf{x}) = \sum_{k=1}^{K_n} w_k \cdot f_{\mathbf{w}_k, \mathbf{w}_{bias,k}}(\mathbf{x})$$

where for $k \in \{1, \dots, K_n\}$

$$f_{\mathbf{w}_k, \mathbf{w}_{bias,k}}(\mathbf{x}) = \frac{1}{(d_1 - \kappa + 1) \cdot (d_2 - \kappa + 1)} \cdot \sum_{\substack{i \in \{1,\dots,d_1-\kappa+1\}, \\ j \in \{1,\dots,d_2-\kappa+1\}}} o_{(i,j),1,k}^{(L)},$$

$$o_{(i,j),s_2,k}^{(r)} = \sigma\left(\sum_{s_1=1}^{2\cdot\kappa^2} w_{1,1,s_1,s_2,k}^{(r)} \cdot o_{(i,j),s_1,k}^{(r-1)} + w_{s_2,k}^{(r)}\right)$$

$((i,j) \in D, s_2 \in \{1, \dots, k_r\}, r \in \{2, \dots, L\})$ and

$$o_{(i,j),s_2,k}^{(1)} = \sigma\left(\sum_{\substack{t_1,t_2 \in \{1,\dots,\kappa\} \\ (i+t_1-1,j+t_2-1) \in D}} w_{t_1,t_2,1,s_2,k}^{(1)} \cdot x_{i+t_1-1,j+t_2-1} + w_{s_2,k}^{(1)}\right)$$

for $(i,j) \in D$ and $s_2 \in \{1, \dots, 2 \cdot \kappa^2\}$.

Following to the proof of Lemma 2 in Drews and Kohler (2022), we get

$$\|(\nabla_{\mathbf{w}} F_n)(\mathbf{w})\|^2$$

$$
\begin{aligned}
&= \sum_{k=1}^{K_n} \left(\frac{2}{n} \sum_{i=1}^{n} (f_{\mathbf{w}}(X_i) - Y_i) \cdot \frac{\partial f_{\mathbf{w}}}{\partial w_k}(X_i) + c_4 \cdot 2 \cdot w_k \right)^2 \\
&\quad + \sum_{k=1}^{K_n} \sum_{r=1}^{L} \sum_{\substack{s_1 \in \{1,\dots,k_{r-1}\}, s_2 \in \{1,\dots,k_r\} \\ t_1, t_2 \in \{1,\dots,M_r\}}} \left(\frac{2}{n} \sum_{i=1}^{n} (f_{\mathbf{w}}(X_i) - Y_i) \cdot \frac{\partial f_{\mathbf{w}}}{\partial w^{(r)}_{t_1,t_2,s_1,s_2,k}}(X_i) \right)^2 \\
&\quad + \sum_{k=1}^{K_n} \sum_{r=1}^{L} \sum_{s_2 \in \{1,\dots,k_r\}} \left(\frac{2}{n} \sum_{i=1}^{n} (f_{\mathbf{w}}(X_i) - Y_i) \cdot \frac{\partial f_{\mathbf{w}}}{\partial w^{(r)}_{s_2,k}}(X_i) \right)^2 \\
&\leq c_8 \cdot \kappa^4 \cdot K_n \cdot L \cdot \max \left(\max_{k,i} \left(\frac{\partial f_{\mathbf{w}}}{\partial w_k(\mathbf{X}_i)} \right)^2, \max_{t_1,t_2,s_1,s_2,k,r,i} \left(\frac{\partial f_{\mathbf{w}}}{\partial w^{(r)}_{t_1,t_2,s_1,s_2,k}}(\mathbf{X}_i) \right)^2, \right. \\
&\quad \left. \max_{s_2,k,r,i} \left(\frac{\partial f_{\mathbf{w}}}{\partial w^{(r)}_{s_2,k}}(\mathbf{X}_i) \right)^2 \right) \cdot \frac{1}{n} \cdot \sum_{i=1}^{n} (f_{\mathbf{w}}(\mathbf{X}_i) - Y_i)^2 \\
&\quad + 8 \cdot c_4^2 \cdot K_n \cdot (\gamma_n^*)^2.
\end{aligned}
$$

Next we calculate the derivatives

$$
\frac{\partial f_{\mathbf{w}}}{\partial w_k}(\mathbf{x}), \quad \frac{\partial f_{\mathbf{w}}}{\partial w^{(r)}_{t_1,t_2,s_1,s_2,k}}(\mathbf{x}) \quad \text{and} \quad \frac{\partial f_{\mathbf{w}}}{\partial w^{(r)}_{s_2,k}}(\mathbf{x}).
$$

We have

$$
\frac{\partial f_{\mathbf{w}}}{\partial w_k}(\mathbf{x}) = f_{\mathbf{w}_k, \mathbf{w}_{bias,k}}(\mathbf{x}).
$$

Furthermore

$$
\frac{\partial f_{\mathbf{w}}}{\partial w^{(r)}_{t_1,t_2,s_1,s_2,k}}(\mathbf{x}) = w_k \cdot \frac{1}{(d_1 - \kappa + 1) \cdot (d_2 - \kappa + 1)} \cdot \sum_{\substack{i \in \{1,\dots,d_1-\kappa+1\}, \\ j \in \{1,\dots,d_2-\kappa+1\}}} \frac{\partial o^{(L)}_{(i,j),1,k}}{\partial w^{(r)}_{t_1,t_2,s_1,s_2,k}}
$$

and

$$
\frac{\partial f_{\mathbf{w}}}{\partial w^{(r)}_{s_2,k}}(\mathbf{x}) = w_k \cdot \frac{1}{(d_1 - \kappa + 1) \cdot (d_2 - \kappa + 1)} \cdot \sum_{\substack{i \in \{1,\dots,d_1-\kappa+1\}, \\ j \in \{1,\dots,d_2-\kappa+1\}}} \frac{\partial o^{(L)}_{(i,j),1,k}}{\partial w^{(r)}_{s_2,k}}.
$$

In the following we calculate the derivatives

$$
\frac{\partial o^{(L)}_{(i,j),1,k}}{\partial w^{(r)}_{t_1,t_2,s_1,s_2,k}}.
$$

In the L–th layer we have

$$
\frac{\partial o^{(L)}_{(i,j),1,k}}{\partial w^{(L)}_{1,1,s_1,1,k}}
$$

$$= o^{(L-1)}_{(i,j),s_1,k} \cdot \sigma' \left(\sum_{s=1}^{2\cdot\kappa^2} w^{(L)}_{1,1,s,1,k} \cdot o^{(L-1)}_{(i,j),s,k} + w^{(L)}_{1,k} \right)$$

for $(i,j) \in D$, $s_1 \in \{1,\dots,2\cdot\kappa^2\}$ and $k \in \{1,\dots,K_n\}$. For $k \in \{1,\dots,K_n\}$, $r \in \{1,\dots,L-1\}$, $s_1 \in \{1,\dots,M_r\}$, $s_2 \in \{1,\dots,M_{r+1}\}$ and $t_1,t_2 \in \{1,\dots,M_r\}$ we get by using the chain rule

$$\begin{aligned}
&\frac{\partial o^{(L)}_{(i,j),1,k}}{\partial w^{(r)}_{t_1,t_2,s_1,s_2,k}} \\
&= \sigma' \left(\sum_{s=1}^{2\cdot\kappa^2} w^{(L)}_{1,1,s,1,k} \cdot o^{(L-1)}_{(i,j),s,k} + w^{(L)}_{1,k} \right) \cdot \sum_{s^{(L)}=1}^{2\cdot\kappa^2} w^{(L)}_{1,1,s^{(L)},1,k} \cdot \frac{\partial o^{(L-1)}_{(i,j),s^{(L)},k}}{\partial w^{(r)}_{t_1,t_2,s_1,s_2,k}} \\
&= \dots \\
&= \sum_{s^{(L)}=1}^{2\cdot\kappa^2} \sum_{s^{(L-1)}=1}^{2\cdot\kappa^2} \cdots \sum_{s^{(r+2)}=1}^{2\cdot\kappa^2} \sigma' \left(\sum_{s=1}^{2\cdot\kappa^2} w^{(L)}_{1,1,s,1,k} \cdot o^{(L-1)}_{(i,j),s,k} + w^{(L)}_{1,k} \right) \cdot w^{(L)}_{1,1,s^{(L)},1,k} \cdot \\
&\quad \sigma' \left(\sum_{s=1}^{2\cdot\kappa^2} w^{(L-1)}_{1,1,s,s^{(L)},k} \cdot o^{(L-2)}_{(i,j),s,k} + w^{(L-1)}_{s^{(L)},k} \right) \cdot w^{(L-1)}_{1,1,s^{(L-1)},s^{(L)},k} \cdots \\
&\quad \sigma' \left(\sum_{s=1}^{2\cdot\kappa^2} w^{(r+1)}_{1,1,s,s^{(r+2)},k} \cdot o^{(r)}_{(i,j),s,k} + w^{(r+1)}_{s^{(r+2)},k} \right) \cdot w^{(r+1)}_{1,1,s_2,s^{(r+2)},k} \cdot \\
&\quad \Bigg(I_{\{r>1\}} \cdot \sigma' \left(\sum_{s=1}^{2\cdot\kappa^2} w^{(r)}_{1,1,s,s_2,k} \cdot o^{(r-1)}_{(i,j),s,k} + w^{(r)}_{s_2,k} \right) \cdot o^{(r-1)}_{(i,j),s_1,k} \\
&\quad + I_{\{r=1\}} \cdot \sigma' \left(\sum_{\substack{\tilde{t}_1,\tilde{t}_2 \in \{1,\dots,\kappa\} \\ (i+\tilde{t}_1-1,j+\tilde{t}_2-1) \in D}} w^{(1)}_{\tilde{t}_1,\tilde{t}_2,1,s_2,k} \cdot x_{i+\tilde{t}_1-1,j+\tilde{t}_2-1} + w^{(1)}_{s_2,k} \right) \cdot x_{i+t_1-1,j+t_2-1} \Bigg),
\end{aligned}$$

where we have set $x_{i,j} = 0$ for $(i,j) \notin D$. For the partial derivatives with respect to $w^{(r)}_{s_2,k}$ we can easily show a similar result.

Using the assumptions of Lemma 3 we can conclude

$$\begin{aligned}
&\max\left(\max_{k,i} \left(\frac{\partial f_{\mathbf{w}}}{\partial w_k(\mathbf{X}_i)} \right)^2, \max_{t_1,t_2,s_1,s_2,k,r,i} \left(\frac{\partial f_{\mathbf{w}}}{\partial w^{(r)}_{t_1,t_2,s_1,s_2,k}}(\mathbf{X}_i) \right)^2, \max_{s_2,k,r,i} \left(\frac{\partial f_{\mathbf{w}}}{\partial w^{(r)}_{s_2,k}}(\mathbf{X}_i) \right)^2 \right) \\
&\leq c_9 \cdot \kappa^{4L} \cdot \max\{\|\sigma'\|_\infty^{2L}, 1\} \cdot \max\{\|\sigma\|_\infty^2, 1\} \cdot B_n^{2L} \cdot (\gamma_n^*)^2.
\end{aligned}$$

Next we show that for any $\mathbf{x} \in [0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}$

$$|f_{\mathbf{w}}(x) - f_{\mathbf{v}}(x)| \leq K_n \cdot \max\{\|\sigma'\|_\infty^L, 1\} \cdot \gamma_n^* \cdot (4\cdot\kappa^4+1)^L \cdot B_n^L \cdot \max\{\|\sigma\|_\infty, 1\} \cdot \|\mathbf{w}-\mathbf{v}\|_\infty.$$

This follows from

$$
\begin{aligned}
&|f_{\mathbf{w}}(\mathbf{x}) - f_{\mathbf{v}}(\mathbf{x})| \\
&= \left| \sum_{k=1}^{K_n} w_k \cdot f_{\mathbf{w}_k,\mathbf{w}_{bias,k}}(\mathbf{x}) - \sum_{k=1}^{K_n} v_k \cdot f_{\mathbf{v}_k,\mathbf{v}_{bias,k}}(\mathbf{x}) \right| \\
&\leq K_n \cdot \max_{k\in\{1,\dots,K_n\}} \left\{ |w_k - v_k| \cdot \|\sigma\|_\infty, \gamma_n^* \cdot |f_{\mathbf{w}_k,\mathbf{w}_{bias,k}}(\mathbf{x}) - f_{\mathbf{v}_k,\mathbf{v}_{bias,k}}(\mathbf{x})| \right\} \\
&\leq K_n \cdot \max_{k\in\{1,\dots,K_n\}} \left\{ |w_k - v_k| \cdot \|\sigma\|_\infty, \gamma_n^* \cdot \max_{\substack{i\in\{1,\dots,d_1-\kappa+1\}\\ j\in\{1,\dots,d_2-\kappa+1\}}} \left| o_{(i,j),1,k}^{(L)} - \bar{o}_{(i,j),1,k}^{(L)} \right| \right\}
\end{aligned}
$$

(where $\bar{o}_{(i,j),s_2,k}^{(r)}$ is defined by replacing in the definition of $o_{(i,j),s_2,k}^{(r)}$ $(\mathbf{w}_k, \mathbf{w}_{bias,k})$ by $(\mathbf{v}_k, \mathbf{v}_{bias,k})$) and that for $r \in \{1,\dots,L\}$ we have

$$
\begin{aligned}
&\left| o_{(i,j),s_2,k}^{(r)} - \bar{o}_{(i,j),s_2,k}^{(r)} \right| \\
&\leq \max\{\|\sigma'\|_\infty^r, 1\} \cdot (4\cdot\kappa^4 + 1)^r \cdot B_n^r \cdot \max\{\|\sigma\|_\infty, 1\} \\
&\quad \cdot \max\left\{ \max_{\substack{\tilde r\in\{1,\dots,L\}\\ \tilde t_1,\tilde t_2,\tilde s_1,\tilde s_2,\tilde k}} \left| w_{\tilde t_1,\tilde t_2,\tilde s_1,\tilde s_2,\tilde k}^{(\tilde r)} - v_{\tilde t_1,\tilde t_2,\tilde s_1,\tilde s_2,\tilde k}^{(\tilde r)} \right|, \max_{\substack{\tilde r\in\{1,\dots,L\}\\ \tilde s_2,\tilde k}} \left| w_{\tilde s_2,\tilde k}^{(\tilde r)} - v_{\tilde s_2,\tilde k}^{(\tilde r)} \right| \right\},
\end{aligned}
$$

which we can easily be shown by induction on r (cf., e.g., proof of Lemma 5 in Kohler and Krzyżak (2021) for a related proof).

This implies

$$
\begin{aligned}
&\frac{1}{n}\sum_{s=1}^n (Y_s - f_{\mathbf{w}}(X_s))^2 \\
&\leq 2\cdot F_n(\mathbf{v}) + \frac{2}{n}\sum_{s=1}^n (f_{\mathbf{v}}(X_s) - f_{\mathbf{w}}(X_s))^2 \\
&\leq 2\cdot F_n(\mathbf{v}) + 2\cdot K_n^2 \cdot \max\{\|\sigma'\|_\infty^{2L}, 1\} \cdot (\gamma_n^*)^2 \cdot (4\cdot\kappa^4+1)^{2L} \cdot B_n^{2L} \\
&\quad \cdot \max\{\|\sigma\|_\infty, 1\}^2 \cdot \frac{2t_n}{L_n} \cdot \max\{F_n(\mathbf{v}), 1\}.
\end{aligned}
$$

The proof is completed by putting together the above results. □

Lemma 4 *Let $\sigma : \mathbb{R} \to \mathbb{R}$ be bounded and differentiable, and assume that its derivative is Lipschitz continuous and bounded. Let $t_n \geq L_n$, $\gamma_n^* \geq 1$, $B_n \geq 1$ and assume*

$$
|\max\{(\mathbf{w}_1)_k, (\mathbf{w}_2)_k\}| \leq \gamma_n^* \quad (k = 1,\dots,K_n), \tag{20}
$$

$$
|\max\{(\mathbf{w}_1)_{s_1,s_2,t_1,t_2,k}^{(r)}, (\mathbf{w}_2)_{s_1,s_2,t_1,t_2,k}^{(r)}\}| \leq B_n \quad \textit{for } r = 2,\dots,L \tag{21}
$$

and

$$
\|\mathbf{w}_2 - \mathbf{v}\|^2 \leq 8\cdot\frac{t_n}{L_n}\cdot\max\{F_n(\mathbf{v}), 1\}. \tag{22}
$$

Assume $X_1, \dots, X_n \in [0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}$ and define F_n by (8), where the parameters $L \in \mathbb{N}$, $M_1, \dots, M_{L+1} \in \mathbb{N}$ and $k_0, \dots, k_L \in \mathbb{N}$ of the convolutional neural network used in (8) *satisfy $L \geq 2$, $M_2 = \dots = M_L = 1$, $M_1 = M_{L+1} = \kappa$, $k_1 = \dots = k_{L-1} = 2 \cdot \kappa^2$ and $k_0 = k_L = 1$.*

Then we have

$$\begin{aligned}
&\|(\nabla_{\mathbf{w}} F_n)(\mathbf{w}_1) - (\nabla_{\mathbf{w}} F_n)(\mathbf{w}_2)\| \\
&\leq c_{10} \cdot \max\{\sqrt{F_n(\mathbf{v})}, 1\} \cdot (\gamma_n^*)^2 \cdot B_n^{3L-1} \cdot K_n^{3/2} \cdot \sqrt{\frac{t_n}{L_n}} \cdot \|\mathbf{w}_1 - \mathbf{w}_2\|.
\end{aligned}$$

Proof. Using the formulas for the partial derivatives derived in the proof of Lemma 3 the assertion follows as in the proof of Lemma 3 in Drews and Kohler (2022). □

Lemma 5 *Let $\alpha \geq 1$, $\beta > 0$ and let $A, B, C \geq 1$. Let $\sigma : \mathbb{R} \to \mathbb{R}$ be k-times differentiable such that all derivatives up to order k are bounded on $\mathbb{R}$. Let $d_1, d_2, \kappa \in \mathbb{N}$ such that $\kappa \leq \min\{d_1, d_2\}$. Let $\mathcal{F}$ be the set of all functions $f_{\mathbf{w}}$ defined by (3)–(6) with*

$$M_1 = M_{L+1} = \kappa, \quad M_2 = \dots = M_L = 1, \quad k_1 = \dots = k_{L-1} = 2 \cdot \kappa^2 \quad \textit{and} \quad k_0 = k_L = 1, \tag{23}$$

where the weights satisfy

$$\sum_{j=1}^{K_n} |w_k| \leq C, \tag{24}$$

$$\max\{|w_{t_1,t_2,s_1,s_2,k}^{(l)}|, |w_{s_2,k}^{(l)}|\} \leq B \quad (k \in \{1, \dots, K_n\}, l \in \{2, \dots, L\}) \tag{25}$$

and

$$\max\{|w_{t_1,t_2,s_1,s_2,k}^{(1)}|, |w_{s_2,k}^{(1)}|\} \leq A \quad (k \in \{1, \dots, K_n\}). \tag{26}$$

Then we have for any $1 \leq p < \infty$, $0 < \epsilon < \beta$ and $\mathbf{x}_1^n \in [0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}$

$$\begin{aligned}
&\mathcal{N}_p\left(\epsilon, \{T_\beta f \,:\, f \in \mathcal{F}\}, \mathbf{x}_1^n\right) \\
&\leq \left(c_{11} \cdot \frac{\beta^p}{\epsilon^p}\right)^{c_{12} \cdot A^{\kappa^2} \cdot B^{(L-1)\cdot\kappa^2} \left(\frac{C}{\epsilon}\right)^{\kappa^2/k} + c_{13}}.
\end{aligned}$$

For the proof of Lemma 5 we need the following result from Kohler and Krzyżak (2022).

Lemma 6 *Let $\alpha \geq 1$, $\beta > 0$ and let $A, B, C \geq 1$. Let $\sigma : \mathbb{R} \to \mathbb{R}$ be k-times differentiable such that all derivatives up to order k are bounded on $\mathbb{R}$. Let $d \in \mathbb{N}$ and let $1 \leq d^* \leq d$. For $x = (x^{(1)}, \dots, x^{(d)})$ and $I \subset \{1, \dots, d\}$ set $x_I = (x^{(i)})_{i \in I}$. Let $\mathcal{F}$ be the set of all functions*

$$f_{\mathbf{w}}(x) = \sum_{I \subset \{1,\dots,d\}, |I| = d^*} f_{\mathbf{w}_I}(x_I)$$

where the $f_{\mathbf{w}_I}(z)$ are defined for $z \in \mathbb{R}^{d^*}$ by

$$f_{\mathbf{w}_I}(z) = \sum_{k=1}^{K_n} (\mathbf{w}_I)_{1,1,k}^{(L)} \cdot f_{\mathbf{w}_I,k,1}^{(L)}(z) \tag{27}$$

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

$$f_{\mathbf{w}_I,k,i}^{(l)}(z) = \sigma\left(\sum_{j=1}^{r} (\mathbf{w}_I)_{k,i,j}^{(l-1)} \cdot f_{\mathbf{w}_I,k,j}^{(l-1)}(z) + (\mathbf{w}_I)_{k,i,0}^{(l-1)}\right) \tag{28}$$

for some $(\mathbf{w}_I)_{k,i,0}^{(l-1)}, \dots, (\mathbf{w}_I)_{k,i,r}^{(l-1)} \in \mathbb{R}$ $(l = 2, \dots, L)$ and

$$f_{\mathbf{w}_I,k,i}^{(1)}(z) = \sigma\left(\sum_{j=1}^{d^*} (\mathbf{w}_I)_{k,i,j}^{(0)} \cdot z^{(j)} + (\mathbf{w}_I)_{k,i,0}^{(0)}\right) \tag{29}$$

for some $(\mathbf{w}_I)_{k,i,0}^{(0)}, \dots, (\mathbf{w}_I)_{k,i,d}^{(0)} \in \mathbb{R}$, and where $\mathbf{w}_I$ denotes the vector of all weights $(\mathbf{w}_I)_{1,1,j}^{(L)}$ and $(\mathbf{w}_I)_{k,i,j}^{(l)}$ $(l = 1, \dots, L-1)$, and where for each $I \subseteq \{1, \dots, d\}$, $|I| = d^*$ the weight vector $\mathbf{w}_I$ satisfies

$$\sum_{j=1}^{K_n} |(\mathbf{w}_I)_{1,1,j}^{(L)}| \le C, \tag{30}$$

$$|(\mathbf{w}_I)_{k,i,j}^{(l)}| \le B \quad (k \in \{1, \dots, K_n\}, i, j \in \{1, \dots, r\}, l \in \{1, \dots, L-1\}) \tag{31}$$

and

$$|(\mathbf{w}_I)_{k,i,j}^{(0)}| \le A \quad (k \in \{1, \dots, K_n\}, i \in \{1, \dots, r\}, j \in \{1, \dots, d^*\}). \tag{32}$$

Then we have for any $1 \le p < \infty$, $0 < \epsilon < \beta$ and $x_1^n \in [-\alpha, \alpha]^d$

$$\begin{aligned}&\mathcal{N}_p\left(\epsilon, \{T_\beta f \,:\, f \in \mathcal{F}\}, x_1^n\right)\\ &\le \left(c_{14} \cdot \frac{\beta^p}{\epsilon^p}\right)^{c_{15} \cdot \alpha^{d^*} \cdot A^{d^*} \cdot B^{(L-1) \cdot d^*} \left(\frac{C}{\epsilon}\right)^{d^*/k} + c_{16}}.\end{aligned}$$

Proof. See Lemma 8 in Kohler and Krzyżak (2022) and its proof. □

Proof of Lemma 5. Set

$$I = \{0, \dots, \kappa - 1\} \times \{0, \dots, \kappa - 1\}.$$

Using (23) it is easy to see that we can find weight vectors $\mathbf{w}_{k,(i,j)}$ such that

$$f_{\mathbf{w}_k, \mathbf{w}_{bias,k}}(\mathbf{x}) = \frac{1}{(d_1 - \kappa + 1) \cdot (d_2 - \kappa + 1)} \cdot \sum_{\substack{i \in \{1, \dots, d_1 - \kappa + 1\}, \\ j \in \{1, \dots, d_2 - \kappa + 1\}}} f_{\mathbf{w}_{k,(i,j)},k,1}^{(L)}(\mathbf{x}_{(i,j)+I})$$

holds. This implies

$$f_{\mathbf{w}}(\mathbf{x}) = \sum_{\substack{i \in \{1,\dots,d_1-\kappa+1\}, \\ j \in \{1,\dots,d_2-\kappa+1\}}} \sum_{k=1}^{K_n} \frac{1}{(d_1-\kappa+1)\cdot(d_2-\kappa+1)} w_k \cdot f^{(L)}_{\mathbf{w}_{k,(i,j)},k,1}(\mathbf{x}_{(i,j)+I}).$$

Furthermore from (24), (25) and (26) we can conclude that (30), (31) and (32) hold. Application of Lemma 6 with $d = d_1 \cdot d_2$ and $d^* = \kappa^2$ yields the desired result. □

Lemma 7 *Let $d_1, d_2, \kappa \in \mathbb{N}$ with $\min\{d_1, d_2\} \geq \kappa$ and set*

$$I = \{0,\dots,\kappa-1\} \times \{0,\dots,\kappa-1\}.$$

Let $1/2 \leq p \leq 1$, $C > 0$, let $f : [0,1]^{(1,1)+I} \to \mathbb{R}$ be a (p, C)–smooth function, let X be $[0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}$-valued random vector and for $x \in [0,1]^{\{1,\dots,d_1\}\times\{1,\dots,d_2\}}$ set

$$m(\mathbf{x}) = \frac{1}{(d_1-\kappa+1)\cdot(d_2-\kappa+1)} \cdot \sum_{(i,j)\in\mathbb{Z}^2 \,:\, (i,j)+I \subseteq \{1,\dots,d_1\}\times\{1,\dots,d_2\}} f\left(\mathbf{x}_{(i,j)+I}\right).$$

Let $l \in \mathbb{N}$, $0 < \delta < 1/2$ with

$$c_{17} \cdot \delta \leq \frac{1}{2^l} \leq c_{18} \cdot \delta \tag{33}$$

and let $L, s \in \mathbb{N}$ with $L \geq 2$, set

$$M_1 = M_{L+1} = \kappa, \quad M_2 = \dots = M_L = 1, \quad k_1 = \dots = k_{L-1} = 2 \cdot \kappa^2 \quad \textit{and} \quad k_0 = k_L = 1,$$

and let

$$\tilde{K}_n \geq \left(l \cdot (2^l+1)^{2\kappa^2} + 1\right)^3.$$

Let

$$f_{\mathbf{w}}(\mathbf{x}) = \sum_{k=1}^{\tilde{K}_n} w_k \cdot f_{\mathbf{w}_k,\mathbf{w}_{bias,k}}(\mathbf{x})$$

where $f_{\mathbf{w}_k,\mathbf{w}_{bias,k}}(\mathbf{x})$ is defined by (4), (5) and (6). Then there exist

$$w_k, w^{(l)}_{t_1,t_2,s_1,s_2,k}, w^{(l)}_{s_2,k} \in [-c_2 \cdot (\log n)^2, c_2 \cdot (\log n)^2] \quad (l = 2,\dots,L, k = 1,\dots,\tilde{K}_n)$$

and

$$w^{(1)}_{t_1,t_2,s_1,s_2,k}, w^{(1)}_{s_2,k} \in \left[-\frac{8\cdot\kappa^2\cdot(\log n)^2}{\delta}, \frac{8\cdot\kappa^2\cdot(\log n)^2}{\delta}\right] \quad (k = 1,\dots,\tilde{K}_n)$$

such that for all $\bar{\mathbf{w}}$ satisfying $|\bar{w}^{(l)}_{t_1,t_2,s_1,s_2,k} - w^{(l)}_{t_1,t_2,s_1,s_2,k}| \leq \log n$ and $|\bar{w}^{(l)}_{s_2,k} - w^{(l)}_{s_2,k}| \leq \log n$ $(l = 1,\dots,L)$ we have for n sufficiently large

$$\int |\sum_{k=1}^{\tilde{K}_n} w_k \cdot f^{(L)}_{\bar{\mathbf{w}}_k,\bar{\mathbf{w}}_{bias,k}}(\mathbf{x}) - m(x)|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x})$$

$$\leq c_{19} \cdot \left(l^2 \cdot (d_1 - \kappa + 1) \cdot (d_2 - \kappa + 1) \cdot \delta + \delta^{2p} + \frac{l \cdot (2^l + 1)^{2 \cdot \kappa^2}}{n^s} \right), \tag{34}$$

$$|\sum_{k=1}^{\tilde{K}_n} w_k \cdot f^{(L)}_{\bar{\mathbf{w}}_k, \bar{\mathbf{w}}_{bias,k}}(\mathbf{x})| \leq c_{20} \cdot \left(1 + \frac{(2^l + 1)^{2 \cdot \kappa^2}}{n^s} \right) \quad (\mathbf{x} \in [0,1]^{\{1,\dots,d_1\} \times \{1,\dots,d_2\}}) \tag{35}$$

and

$$\sum_{k=1}^{\tilde{K}_n} |w_k|^2 \leq \frac{c_{21}}{2^{2 \cdot \kappa^2 \cdot l}}. \tag{36}$$

To prove Lemma 7 we need the following result from Kohler and Krzyżak (2022).

Lemma 8 *Let $1/2 \leq p \leq 1$, $C > 0$, let $f : \mathbb{R}^d \to \mathbb{R}$ be a (p, C)–smooth function, let $N \in \mathbb{N}$ and let Z_1, ..., Z_N be $[0,1]^d$-valued random vectors. Let $l \in \mathbb{N}$, $0 < \delta < 1/2$ with*

$$c_{22} \cdot \delta \leq \frac{1}{2^l} \leq c_{23} \cdot \delta \tag{37}$$

and let $L, r, s \in \mathbb{N}$ with

$$L \geq 2 \quad \textit{and} \quad r \geq 2d$$

and let

$$\tilde{K}_n \geq \left(l \cdot (2^l + 1)^{2d} + 1 \right)^3$$

Define $f^{(L)}_{\mathbf{w},k,1}$ by (28) and (29) with d^* replaced by d. Then there exist

$$\mathbf{w}^{(l)}_{k,i,j} \in [-c_2 \cdot (\log n)^2, c_2 \cdot (\log n)^2] \quad (l = 1, \dots, L, k = 1, \dots \tilde{K}_n)$$

and

$$\mathbf{w}^{(0)}_{k,i,j} \in \left[-\frac{8 \cdot d \cdot (\log n)^2}{\delta}, \frac{8 \cdot d \cdot (\log n)^2}{\delta} \right] \quad (k = 1, \dots, \tilde{K}_n).$$

such that for all $\bar{\mathbf{w}}$ satisfying $|\bar{w}^{(l)}_{i,j,k} - \mathbf{w}^{(l)}_{i,j,k}| \leq \log n$ $(l = 0, \dots, L-1)$ we have for n sufficiently large

$$\max_{i=1,\dots,N} \int |\sum_{k=1}^{\tilde{K}_n} w^{(L)}_{1,1,k} \cdot f^{(L)}_{\bar{\mathbf{w}},k,1}(x) - f(x)|^2 \mathbf{P}_{Z_i}(dx)$$
$$\leq c_{24} \cdot \left(l^2 \cdot N \cdot \delta + \delta^{2p} + \frac{l \cdot (2^l + 1)^{2d}}{n^s} \right), \tag{38}$$

$$|\sum_{k=1}^{\tilde{K}_n} w^{(L)}_{1,1,k} \cdot f^{(L)}_{\bar{\mathbf{w}},k,1}(x)| \leq c_{25} \cdot \left(1 + \frac{(2^l + 1)^{2d}}{n^s} \right) \quad (x \in [0,1]^d) \tag{39}$$

and

$$\sum_{k=1}^{\tilde{K}_n} |w^{(L)}_{1,1,k}|^2 \leq \frac{c_{26}}{2^{2 \cdot d \cdot l}}. \tag{40}$$

Proof. The result follows by an easy modification of the proof of Lemma 7 in Kohler and Krzyżak (2022). The only difference in the proof is that at the very beginning we use a sequence of coverings $\mathcal{P}^{(0)} = \{[0,1]^d\}$, $\mathcal{P}^{(1)}$, ..., $\mathcal{P}^{(l)}$ of $[0,1]^d$ with the following properties:

1. $\mathcal{P}^{(k)}$ consists of $(2^k+1)^d$ many pairwise disjoint cubes of side length $1/2^k$ ($k = 1, \dots, l$).

2. $[0,1]^d \subseteq \cup_{A \in \mathcal{P}^{(k)}} A$

3. $$\sum_{i=1}^N \mathbf{P}_{Z_i} \left(\cup_{A \in \mathcal{P}^{(k)}} A_{border,\delta} \right) \leq 4d \cdot 2^k \cdot N \cdot \delta, \tag{41}$$

   where

   $$\begin{aligned} A_{border,\delta} &= [u^{(1)} - \delta, v^{(1)} + \delta] \times \cdots \times [u^{(d)} - \delta, v^{(d)} + \delta] \\ &\qquad \setminus [u^{(1)} + \delta, v^{(1)} - \delta] \times \cdots \times [u^{(d)} + \delta, v^{(d)} - \delta] \end{aligned}$$

   for

   $$A = [u^{(1)}, v^{(1)}] \times \cdots \times [u^{(d)}, v^{(d)}].$$

We can ensure (41) by shifting a partition of

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

consisting of $(2^k+1)^d$ many cubes of side length $1/2^k$ separately in each component by multiples of $2 \cdot \delta$ less than or equal to $1/2^k$, which gives us for each component

$$\left\lfloor \frac{1}{2 \cdot \delta} \cdot \frac{1}{2^k} \right\rfloor$$

disjoint sets of which at least one must have $\sum_{i=1}^N \mathbf{P}_{Z_i}$-measure less than or equal to

$$\frac{N}{\lfloor \frac{1}{2 \cdot \delta} \cdot \frac{1}{2^k} \rfloor} \leq \frac{N}{\frac{1}{2 \cdot \delta} \cdot \frac{1}{2^k} - 1} \leq N \cdot \frac{2 \cdot \delta \cdot 2^k}{1 - 2 \cdot \delta \cdot 2^k} \leq 4 \cdot N \cdot \delta \cdot 2^k$$

in case $2 \cdot \delta \cdot 2^k \leq 1/2$, which we can assume w.l.o.g. (because otherwise (41) is always satisfied).

From this we get the assertion as in the proof of Lemma 7 in Kohler and Krzyżak (2022). □

Proof of Lemma 7. We apply Lemma 8 with $d = \kappa^2$ and $N = (d_1 - \kappa + 1) \cdot (d_2 - \kappa + 1)$ and choose $w_k, w_{t_1,t_2,s_1,s_2,k}^{(l)}, w_{s_2,k}^{(l)}$ such that we have

$$o_{(i,j),1,k}^{(L)} = f_{\mathbf{w},k,1}^{(L)}(\mathbf{x}_{(i,j)+I}) \quad \text{for all } (i,j) \in \{1, \dots, d_1 - \kappa + 1\} \times \{1, \dots, d_2 - \kappa + 1\}$$

and

$$w_k = w_{1,1,k}^{(L)} \quad (k = 1, \dots, \tilde{K}_n).$$

This implies

$$\int |\sum_{k=1}^{\tilde{K}_n} w_k \cdot f_{\bar{\mathbf{w}}_k, \bar{\mathbf{w}}_{bias,k}}^{(L)}(\mathbf{x}) - m(x)|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x})$$

$$\leq \frac{1}{(d_1 - \kappa + 1) \cdot (d_2 - \kappa + 1)} \cdot \sum_{\substack{i \in \{1, \dots, d_1 - \kappa + 1\}, \\ j \in \{1, \dots, d_2 - \kappa + 1\}}} \int \left| \sum_{k=1}^{\tilde{K}_n} w_{1,1,k}^{(L)} \cdot f_{\bar{\mathbf{w}},k,1}^{(L)}(\mathbf{x}_{(i,j)+I}) - f\left(\mathbf{x}_{(i,j)+I}\right) \right|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x})$$

from which we get the assertion by Lemma 8. □

In order to be able to formulate our next auxiliary result we need the following notation: Let $(x_1, y_1), \dots, (x_n, y_n) \in \mathbb{R}^d \times \mathbb{R}$, let $K \in \mathbb{N}$, let $B_1, \dots, B_K : \mathbb{R}^d \to \mathbb{R}$ and let $c_{27} > 0$. In the next lemma we consider the problem to minimize

$$F(\mathbf{a}) = \frac{1}{n} \sum_{i=1}^{n} |\sum_{k=1}^{K} a_k \cdot B_k(x_i) - y_i|^2 + c_{27} \cdot \sum_{k=1}^{K_n} a_k^2, \tag{42}$$

where $\mathbf{a} = (a_1, \dots, a_K)^T$, by gradient descent. To do this, we choose $\mathbf{a}^{(0)} \in \mathbb{R}^K$ and set

$$\mathbf{a}^{(t+1)} = \mathbf{a}^{(t)} - \lambda_n \cdot (\nabla_{\mathbf{a}} F)(\mathbf{a}^{(t)}) \tag{43}$$

for some properly chosen $\lambda_n > 0$.

Lemma 9 *Let F be defined by (42) and choose $\mathbf{a}_{opt}$ such that*

$$F(\mathbf{a}_{opt}) = \min_{\mathbf{a} \in \mathbb{R}^K} F(\mathbf{a}).$$

Then for any $\mathbf{a} \in \mathbb{R}^K$ we have

$$\|(\nabla_{\mathbf{a}} F)(\mathbf{a})\|^2 \geq 4 \cdot c_{27} \cdot (F(\mathbf{a}) - F(\mathbf{a}_{opt})).$$

Proof. See Lemma 8 in Drews and Kohler (2022). □

4.2 Proof of Theorem 1

The result follows by a more or less straightforward modification of the proof of Theorem 1 in Kohler and Krzyżak (2022) using Lemma 5 and Lemma 7 instead of Lemma 6 and Lemma 8. For the sake of completeness we nevertheless present a complete proof.

For $z \in \mathbb{R}$ we have that $T_1 z > 1/2$ holds if and only if $z > 1/2$, hence we can assume w.l.o.g. that our estimate is given by

$$f_n(\mathbf{x}) = \begin{cases} 1, & \text{if } m_n(\mathbf{x}) \geq \frac{1}{2}, \\ 0, & \text{elsewhere} \end{cases}$$

where

$$m_n(\mathbf{x}) = T_1 \left(f_{\mathbf{w}^{(t_n)}}(\mathbf{x}) \right).$$

Consequently we know by Lemma 1 that it suffices to show that we have for any $\epsilon > 0$

$$\mathbf{E} \int |m_n(\mathbf{x}) - \eta(\mathbf{x})|^2 \mathbf{P}_X(d\mathbf{x}) \leq c_{28} \cdot n^{-\frac{1}{\kappa^2+1}+\epsilon} \tag{44}$$

Set $I = \{1, \ldots, \kappa\} \times \{1, \ldots, \kappa\}$, $r = (d_1 - \kappa + 1) \cdot (d_2 - \kappa + 1)$, $\delta = c_{29} \cdot n^{-1/(1+\cdot\kappa^2)}$ and

$$\tilde{K}_n = n^6.$$

Using Lemma 7 (with $\delta = n^{-1/(1+\kappa^2)}$ and sufficiently large s) we can construct a weight vector $\mathbf{w}$ of a convolutional neural network

$$\tilde{f}_{\mathbf{w}}(\mathbf{x}) = \sum_{k=1}^{\tilde{K}_n} w_k \cdot f_{\mathbf{w}_k, \mathbf{w}_{bias,k}}(\mathbf{x})$$

with the property that for any weight vector $\bar{\mathbf{w}}$ with

$$|\bar{w}^{(r)}_{t_1,t_2,s_1,s_2,k} - w^{(r)}_{t_1,t_2,s_1,s_2,k}| \leq \log n \quad \text{and} \quad |\bar{w}^{(r)}_{s_2,k} - w^{(r)}_{s_2,k}| \leq \log n$$

$(r = 1, \ldots, L)$ we have

$$\int \left| \sum_{k=1}^{\tilde{K}_n} w_k \cdot f_{\bar{\mathbf{w}}_k, \bar{\mathbf{w}}_{bias,k}}(\mathbf{x}) - \frac{\sum_{(i,j)\in\{1,\ldots,d_1-\kappa+1\}\times\{1,\ldots,d_2-\kappa+1\}} f(\mathbf{x}_{(i,j)+I})}{(d_1-\kappa+1)\cdot(d_2-\kappa+1)} \right|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x})$$
$$\leq c_{30} \cdot (\log n)^2 \cdot n^{-\frac{1}{1+\kappa^2}} \tag{45}$$

and

$$\sum_{k=1}^{\tilde{K}_n} w_k^2 \leq \frac{c_{31}}{n}.$$

Let A_n be the event that there exists pairwise distinct $j_1, \ldots, j_{\tilde{K}_n} \in \{1, \ldots, K_n\}$ such that the randomly initialized weights satisfy

$$|(\mathbf{w}^{(0)})^{(r)}_{t_1,t_2,s_1,s_2,j_k} - w^{(r)}_{t_1,t_2,s_1,s_2,k}| \leq \log n \tag{46}$$

and

$$|(\mathbf{w}^{(0)})^{(r)}_{s_2,j_k} - w^{(r)}_{s_2,k}| \leq \log n \tag{47}$$

for $r = 1, \dots, L$ and $k = 1, \dots, \tilde{K}_n$.

Define the weight vectors $(\mathbf{w}^*)^{(t)}$ $(t = 0, 1, \dots, t_n)$ by

$$((\mathbf{w}^*)^{(t)})_{j_k} = w_k \quad \text{if } k \in \{1, \dots, \tilde{K}_n\},$$

$$((\mathbf{w}^*)^{(t)})_k = 0 \quad \text{if } k \notin \{j_1, \dots, j_{\tilde{K}_n}\},$$

$$((\mathbf{w}^*)^{(t)})^{(l)}_{s_1,s_2,t_1,t_2,k} = (\mathbf{w}^{(t)})^{(l)}_{s_1,s_2,t_1,t_2,k} \quad \text{if } l \in \{1, \dots, L\}$$

and

$$((\mathbf{w}^*)^{(t)})^{(l)}_{s_2,k} = (\mathbf{w}^{(t)})^{(l)}_{s_2,k} \quad \text{if } l \in \{1, \dots, L\}.$$

In order to show (44) we will use the following error decomposition:

$$\begin{aligned}
&\int |m_n(\mathbf{x}) - \eta(\mathbf{x})|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x}) \\
&= \left(\mathbf{E}\left\{|m_n(\mathbf{X}) - Y|^2 | \mathcal{D}_n\right\} - \mathbf{E}\{|\eta(\mathbf{X}) - Y|^2\}\right) \cdot 1_{A_n} + \int |m_n(\mathbf{x}) - \eta(\mathbf{x})|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x}) \cdot 1_{A_n^c} \\
&= \Big[\mathbf{E}\left\{|m_n(\mathbf{X}) - Y|^2 | \mathcal{D}_n\right\} - \mathbf{E}\{|\eta(\mathbf{X}) - Y|^2\} \\
&\qquad\qquad - \left(2 \cdot \frac{1}{n}\sum_{i=1}^n |m_n(\mathbf{X}_i) - Y_i|^2 - 2 \cdot \frac{1}{n}\sum_{i=1}^n |\eta(\mathbf{X}_i) - Y_i|^2\right)\Big] \cdot 1_{A_n} \\
&\quad + \Big[2 \cdot \frac{1}{n}\sum_{i=1}^n |m_n(\mathbf{X}_i) - Y_i|^2 - 2 \cdot \frac{1}{n}\sum_{i=1}^n |\eta(\mathbf{X}_i) - Y_i|^2\Big] \cdot 1_{A_n} \\
&\quad + \int |m_n(\mathbf{x}) - \eta(\mathbf{x})|^2 \mathbf{P}_{\mathbf{X}}(dx) \cdot 1_{A_n^c} \\
&=: \sum_{j=1}^3 T_{j,n}.
\end{aligned}$$

In the reminder of the proof we bound

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

for $j \in \{1, 2, 3\}$.

In the *first step of the proof* we show

$$\mathbf{E} T_{3,n} \leq \frac{c_{32}}{n}.$$

The definition of m_n implies $\int |m_n(\mathbf{x}) - \eta(\mathbf{x})|^2 \mathbf{P}_{\mathbf{X}}(d\mathbf{x}) \leq 4$, hence it suffices to show

$$\mathbf{P}(A_n^c) \leq \frac{c_{33}}{n}. \tag{48}$$

To do this, we consider sequential choice of the initial weights of the K_n convolutional neural networks which we compute in parallel.

Probability that the weights in the first of these networks differ in all components by at most $\log n$ from $w_{t_1,t_2,s_1,s_2,1}^{(l)}$, $w_{s_2,1}^{(l)}(l=1,\dots,L)$ is for large n bounded below by

$$\left(\frac{\log n}{2\cdot c_2\cdot(\log n)^2}\right)^{2\cdot\kappa^2\cdot(2\cdot\kappa^2+1)\cdot(L-1)}\cdot\left(\frac{\log n}{2\cdot c_3\cdot n^\tau}\right)^{2\cdot\kappa^2\cdot(\kappa^2+1)} \quad\geq\quad n^{-2\cdot\kappa^2\cdot(\kappa^2+1)\cdot\tau-0.5}.$$

Hence probability that none of the first $n^{2\cdot\kappa^2\cdot(\kappa^2+1)\cdot\tau+1}$ neural networks satisfies this condition is for large n bounded above by

$$\begin{aligned}(1-n^{-2\cdot\kappa^2\cdot(\kappa^2+1)\cdot\tau-0.5})^{n^{2\cdot\kappa^2\cdot(\kappa^2+1)\cdot\tau+1}} &\leq \left(\exp\left(-n^{-2\cdot\kappa^2\cdot(\kappa^2+1)\cdot\tau-0.5}\right)\right)^{n^{2\cdot\kappa^2\cdot(\kappa^2+1)\cdot\tau+1}}\\ &= \exp(-n^{0.5}).\end{aligned}$$

Since we have $K_n\geq n^{2\cdot\kappa^2\cdot(\kappa^2+1)\cdot\tau+1}\cdot\tilde{K}_n$ for large n we can successively use the same construction for all of $\tilde{K}_n$ weights and we can conclude: Probability that there exists $k\in\{1,\dots,\tilde{K}_n\}$ such that none of the K_n weight vectors of the convolutional neural network differs by at most $\log n$ from $w_{t_1,t_2,s_1,s_2,k}^{(l)}$, $w_{s_2,k}^{(l)}$ is for large n bounded from above by

$$\mathbf{P}(A_n^c)=\tilde{K}_n\cdot\exp(-n^{0.5})\leq n^\rho\cdot\exp(-n^{0.5})\leq\frac{c_{33}}{n}.$$

In the *second step of the proof* we show for large n

$$\|\mathbf{w}^{(t)}-\mathbf{w}^{(0)}\|\leq\log n \tag{49}$$

for all $t=1,\dots,t_n$. For large n we have

$$F_n(\mathbf{w}^{(0)})=\frac{1}{n}\sum_{i=1}^n|Y_i-0|^2+0\leq 1$$

and

$$2\cdot\frac{t_n}{L_n}\leq(\log n)^2.$$

Application of Lemma 3 and Lemma 4 with $\gamma_n^*=\log n$ and $B_n=(c_2+1)\cdot(\log n)^2$ yields that the assumptions (14) and (15) of Lemma 2 are satisfied. Lemma 2 implies the assertion.

Let $\epsilon>0$ be arbitrary. In the *third step of the proof* we show

$$\mathbf{E}T_{1,n}\leq c_{34}\cdot\frac{n^{\tau\cdot\kappa^2+\epsilon}}{n}.$$

Let $\mathcal{W}_n$ be the set of all weight vectors $\mathbf{w}$ which satisfy

$$|w_k|\leq(\log n)^2\quad(k=1,\dots,K_n),$$

$$\max\left\{|w_{t_1,t_2,s_1,s_2,k}^{(l)}|,|w_{s_2,k}^{(l)}|\right\}\leq(c_2+1)\cdot(\log n)^2\quad(l=2,\dots,L)$$

and

$$\max\left\{|w^{(1)}_{t_1,t_2,s_1,s_2,k}|, |w^{(1)}_{s_2,k}|\right\} \leq (c_3+1)\cdot(\log n)^2\cdot n^{\tau}.$$

By the second step and the initial choice of $\mathbf{w}^{(0)}$ we know that on A_n we have

$$\mathbf{w}^{(t)} \in \mathcal{W}_n \quad (t=0,\dots,t_n).$$

Hence, for any $u>0$ we get for large n

$$\begin{aligned}
&\mathbf{P}\{T_{1,n}>u\}\\
&\leq \mathbf{P}\Bigg\{\exists f\in\mathcal{F}_n : \mathbf{E}\left(|f(\mathbf{X})-Y|^2\right)-\mathbf{E}\left(|\eta(\mathbf{X})-Y|^2\right)\\
&\qquad\qquad -\frac{1}{n}\sum_{i=1}^{n}\left(|f(\mathbf{X}_i)-Y_i|^2-|\eta(\mathbf{X}_i)-Y_i|^2\right)\\
&\qquad\qquad > \frac{1}{2}\cdot\left(u+\mathbf{E}\left(|f(\mathbf{X})-Y|^2\right)-\mathbf{E}\left(|\eta(\mathbf{X})-Y|^2\right)\right)\Bigg\},
\end{aligned}$$

where

$$\mathcal{F}_n = \left\{T_1(f_{\mathbf{w}}) \quad : \quad \mathbf{w}\in\mathcal{W}_n\right\}.$$

By Lemma 5 we have

$$\mathcal{N}_1\left(\delta,\mathcal{F}_n,x_1^n\right) \leq \left(\frac{c_{34}}{\delta}\right)^{c_{35}\cdot(\log n)^{2\kappa^2} n^{\tau\cdot\kappa^2}\cdot(\log n)^{2\cdot(L-1)\cdot\kappa^2}\cdot\left(\frac{K_n\cdot(\log n)^2}{\delta}\right)^{\kappa^2/k}+c_{36}}.$$

By choosing k large enough we get for $\delta>1/n^2$

$$\mathcal{N}_1\left(\delta,\mathcal{F}_n,x_1^n\right) \leq c_{37}\cdot n^{c_{38}\cdot n^{\tau\cdot\kappa^2+\epsilon/2}}.$$

This together with Theorem 11.4 in Györfi et al. (2002) leads for $u\geq 1/n$ to

$$\mathbf{P}\{T_{1,n}>u\} \leq 14\cdot c_{37}\cdot n^{c_{38}\cdot n^{\tau\cdot\kappa^2+\epsilon/2}}\cdot\exp\left(-\frac{n}{5136}\cdot u\right).$$

For $\epsilon_n\geq 1/n$ we can conclude for large n

$$\begin{aligned}
\mathbf{E}\{T_{1,n}\} &\leq \epsilon_n+\int_{\epsilon_n}^{\infty}\mathbf{P}\{T_{1,n}>u\}\,du\\
&\leq \epsilon_n+14\cdot c_{37}\cdot n^{c_{38}\cdot n^{\tau\cdot\kappa^2+\epsilon/2}}\cdot\exp\left(-\frac{n}{5136}\cdot\epsilon_n\right)\cdot\frac{5136}{n}.
\end{aligned}$$

Setting

$$\epsilon_n=\frac{5136}{n}\cdot c_{38}\cdot n^{\tau\cdot\kappa^2+\epsilon/2}\cdot\log n$$

yields the assertion of the third step of the proof.

In the *fourth step of the proof* we show

$$\mathbf{E}\{T_{2,n}\} \leq c_{39} \cdot \left(\frac{n^{\tau \cdot \kappa^2 + \epsilon}}{n} + n^{-\frac{1}{1+\kappa^2}} \right).$$

Using

$$|T_1(z) - y| \leq |z - y| \quad \text{for } |y| \leq 1$$

we get

$$\begin{aligned} &T_{2,n}/2 \\ &= \Big[\frac{1}{n} \sum_{i=1}^n |m_n(\mathbf{X}_i) - Y_i|^2 - \frac{1}{n} \sum_{i=1}^n |\eta(\mathbf{X}_i) - Y_i|^2 \Big] \cdot 1_{A_n} \\ &\leq \Big[\frac{1}{n} \sum_{i=1}^n |f_{\mathbf{w}^{(t_n)}}(\mathbf{X}_i) - Y_i|^2 - \frac{1}{n} \sum_{i=1}^n |\eta(\mathbf{X}_i) - Y_i|^2 \Big] \cdot 1_{A_n} \\ &\leq \Big[F_n(\mathbf{w}^{(t_n)}) - \frac{1}{n} \sum_{i=1}^n |\eta(\mathbf{X}_i) - Y_i|^2 \Big] \cdot 1_{A_n}. \end{aligned}$$

Application of Lemma 2 (which is possible due to Lemma 3 and Lemma 4) implies that this in turn is less than

$$\Big[F_n(\mathbf{w}^{(t_n-1)}) - \frac{1}{2L_n} \cdot \|\nabla_{\mathbf{w}} F_n(\mathbf{w}^{(t_n-1)})\|^2 - \frac{1}{n} \sum_{i=1}^n |\eta(X_i) - Y_i|^2 \Big] \cdot 1_{A_n}.$$

Since the sum of squares of all partial derivatives is at least as large as the sum of squares of the partial derivatives with respect to the outer weights w_k $(k = 1, \dots, K_n)$, we can upper bound this in turn following Lemma 9 by

$$\begin{aligned} &\Big[F_n(\mathbf{w}^{(t_n-1)}) - \frac{1}{2L_n} \cdot 4 \cdot c_4 \cdot (F_n(\mathbf{w}^{(t_n-1)}) - F_n((\mathbf{w}^*)^{(t_n-1)}) \\ &\qquad\qquad - \frac{1}{n} \sum_{i=1}^n |\eta(\mathbf{X}_i) - Y_i|^2 \Big] \cdot 1_{A_n} \\ &= \Big[\left(1 - \frac{2 \cdot c_4}{L_n}\right) \cdot F_n(\mathbf{w}^{(t_n-1)}) + \frac{2 \cdot c_4}{L_n} \cdot F_n((\mathbf{w}^*)^{(t_n-1)}) - \frac{1}{n} \sum_{i=1}^n |\eta(\mathbf{X}_i) - Y_i|^2 \Big] \cdot 1_{A_n}. \end{aligned}$$

Applying this argument repeatedly shows that

$$\begin{aligned} &T_{2,n}/2 \\ &\leq \Big[\left(1 - \frac{2 \cdot c_4}{L_n}\right)^{t_n} \cdot F_n(\mathbf{w}^{(0)}) + \sum_{k=1}^{t_n} \frac{2 \cdot c_4}{L_n} \cdot \left(1 - \frac{2 \cdot c_4}{L_n}\right)^{k-1} F_n((\mathbf{w}^*)^{(t_n-k)}) \\ &\qquad\qquad - \frac{1}{n} \sum_{i=1}^n |\eta(\mathbf{X}_i) - Y_i|^2 \Big] \cdot 1_{A_n}. \end{aligned}$$

This implies

$$\begin{aligned}
&\mathbf{E}\{T_{2,n}/2\} \\
&\leq \left(1-\frac{2\cdot c_4}{L_n}\right)^{t_n}\cdot\mathbf{E}\{Y^2\}+\sum_{k=1}^{t_n}\frac{2\cdot c_4}{L_n}\cdot\left(1-\frac{2\cdot c_4}{L_n}\right)^{k-1}\cdot \\
&\quad \mathbf{E}\Bigg(\left(\frac{1}{n}\sum_{i=1}^n|f_{(\mathbf{w}^*)^{(t_n-k)}}(\mathbf{X}_i)-Y_i|^2-\frac{1}{n}\sum_{i=1}^n|\eta(\mathbf{X}_i)-Y_i|^2\right)\cdot 1_{A_n}\Bigg) \\
&\quad +c_4\cdot\sum_{k=1}^{\tilde{K}_n}|w_k|^2 \\
&\leq \left(1-\frac{2\cdot c_4}{L_n}\right)^{t_n}\cdot\mathbf{E}\{Y^2\}+c_4\cdot\sum_{k=1}^{\tilde{K}_n}|w_k|^2 \\
&\quad +\sum_{k=1}^{t_n}\frac{2\cdot c_4}{L_n}\cdot\left(1-\frac{2\cdot c_4}{L_n}\right)^{k-1}\cdot 2\cdot \\
&\qquad \mathbf{E}\left(\max_{k=0,\dots,t_n-1}\int|f_{(\mathbf{w}^*)^{(t_n-k)}}(\mathbf{x})-\eta(x)|^2\mathbf{P}_X(d\mathbf{x})\right) \\
&\quad +\sum_{k=1}^{t_n}\frac{2\cdot c_4}{L_n}\cdot\left(1-\frac{2\cdot c_4}{L_n}\right)^{k-1}\cdot \\
&\quad \mathbf{E}\Bigg(\Bigg(\frac{1}{n}\sum_{i=1}^n|f_{(\mathbf{w}^*)^{(t_n-k)}}(\mathbf{X}_i)-Y_i|^2-\frac{1}{n}\sum_{i=1}^n|\eta(\mathbf{X}_i)-Y_i|^2 \\
&\quad -2\cdot\Bigg(\mathbf{E}\{|f_{(\mathbf{w}^*)^{(t_n-k)}}(\mathbf{X})-Y|^2|\mathcal{D}_n,\mathbf{w}^{(0)}\} \\
&\qquad\qquad -\mathbf{E}\{|\eta(\mathbf{X})-Y|^2\}\Bigg)\Bigg)\cdot 1_{A_n}\Bigg).
\end{aligned}$$

Arguing as in the third step of the proof (which is possible even if we do not have truncated functions because of (49) and (35)) we get

$$\begin{aligned}
&\mathbf{E}\Bigg(\Bigg(\frac{1}{n}\sum_{i=1}^n|f_{(\mathbf{w}^*)^{(t_n-k)}}(\mathbf{X}_i)-Y_i|^2-\frac{1}{n}\sum_{i=1}^n|\eta(\mathbf{X}_i)-Y_i|^2 \\
&\quad -2\cdot\Bigg(\mathbf{E}\{|f_{(\mathbf{w}^*)^{(t_n-k)}}(\mathbf{X})-Y|^2|\mathcal{D}_n,\mathbf{w}^{(0)}\} \\
&\qquad\qquad -\mathbf{E}\{|\eta(\mathbf{X})-Y|^2\}\Bigg)\Bigg)\cdot 1_{A_n}\Bigg) \\
&\leq c_{40}\cdot\frac{(\log n)^2}{n}+c_{41}\cdot\frac{n^{\tau\cdot\kappa^2+\epsilon}}{n}.
\end{aligned}$$

From this we conclude

$$
\begin{aligned}
&\mathbf{E}\{T_{2,n}/2\}\\
&\leq \left(1-\frac{2\cdot c_4}{L_n}\right)^{t_n}\cdot \mathbf{E}\{Y^2\}\\
&\quad +2\cdot \mathbf{E}\left(\max_{k=0,\dots,t_n-1}\int |\tilde{f}_{(\mathbf{w}^*)^{(t_n-k)}}(\mathbf{x})-\eta(x)|^2 \mathbf{P}_X(d\mathbf{x})\right)\\
&\quad +c_4\cdot \sum_{k=1}^{\tilde{K}_n}|w_k|^2+c_{40}\cdot\frac{(\log n)^2}{n}+c_{41}\cdot\frac{n^{\tau\cdot\kappa^2+\epsilon}}{n}.
\end{aligned}
$$

The definition of t_n together with $c_5 \geq 1/(2\cdot c_4)$ implies

$$
\begin{aligned}
\left(1-\frac{2\cdot c_4}{L_n}\right)^{t_n}\cdot \mathbf{E}\{Y^2\} &\leq \exp\left(-\frac{2\cdot c_4}{L_n}\cdot t_n\right)\cdot \mathbf{E}\{Y^2\}\\
&\leq \exp(-2\cdot c_4\cdot c_5\cdot \log n)\cdot \mathbf{E}\{Y^2\}\\
&\leq \frac{c_{42}}{n}.
\end{aligned}
$$

And by (45) we know

$$
\max_{k=0,\dots,t_n-1}\int |\tilde{f}_{(\mathbf{w}^*)^{(t_n-k)}}(\mathbf{x})-\eta(x)|^2\mathbf{P}_X(d\mathbf{x}) \leq c_{43}\cdot(\log n)^2\cdot n^{-\frac{1}{1+\kappa^2}}
$$

All the results above imply the assertion.

□

5 Acknowledgment

The first and the third author would like to thank the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) for funding this project (Projektnummer 449102119).

References

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

[2] Arora, S., Cohen, N., Golowich, N., and Hu, W. (2018). A convergence analysis of gradient descent for deep linear neural networks. *International Conference on Learning Representations (ICLR 2019).* New Orleans, Louisiana.

[3] Barron, A. R. (1993). Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory* **39**, pp. 930–944.

[4] Bartlett, P. L., Long, P. M., and Lugosi, G. (2020). Beningn overfitting in linear regression. *Proceedings of the National Academy of Sciences*, **117**, pp. 30063-30070.

[5] Bartlett, P. L., Montanari, A., and Rakhlin, A. (2021). Deep learning: a statistical viewpoint. *arXiv: 2103.09177v1.*

[6] Belkin, M., Rakhlin, A., and Tsybakov, A. B. (2019). Does data interpolation contradict statistical optimality? *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1611-1619.

[7] Berner, J., Grohs, P., Kutyniok, G., and Petersen, P. (2021). The modern mathematics of deep learning. *arXiv: 2105.04026v1.*

[8] Braun, A., Kohler, M., Langer, S., and Walk, H. (2021). The smoking gun: statistical theory improves neural network estimates. Preprint, arXiv: 2107.09550.

[9] Choromanska, A., Henaff, M., Mathieu, M., Arous, G. B., and LeCun, Y. (2015). The loss surface of multilayer networks. International Conference on Articial Intelligence and Statistics (AISTATS) 2015, San Diego, CA, USA. Proceeding of Machine Learning Research, volume 38, pp. 192-204.

[10] Cover, T. M. (1968). Rates of convergence of nearest neighbor procedures. In *Proceedings of the Hawaii International Conference on Systems Sciences*, pp. 413-415, Honolulu, HI.

[11] Devroye, L., Györfi, L., and Lugosi, G. (1996). *A Probabilistic Theory of Pattern Recognition.* Springed, New York, USA.

[12] Devroye, L., and Wagner, T. J. (1980). Distribution-free consistency results in nonparametric discrimination and regression function estimation. *Annals of Statistics*, **8**, pp. 231-239.

[13] Drews, S. and Kohler, M. (2022). On the universal consistency of an over-parametrized deep neural network estimate learned by gradient descent. arXiv:2208.14283.

[14] Du, S., and Lee, J. (2018). On the power of over-parametrization in neural networks with quadratic activation. *Proceedings of the 35th International Conference on Machine Learning (PMLR 2018)*, **80**, pp. 1329-1338. Stockholm, Sweden.

[15] Du, S., Lee, J., Tian, Y., Poczos, B., and Singh, A. (2018). Gradient descent learns one-hidden-layer CNN: don't be afraid of spurious local minima. *Proceedings of the 35th International Conference on Machine Learning (PMLR 2018)*, **80**, pp. 1339-1348. Stockholm, Sweden.

[16] Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning.* MIT Press.

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

[18] Huang, W., Du, W., and Xu, Y.D. (2021). On the neural tangent kernel of deep networks with orthogonal initialization. *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21).* pp. 2577-2583. Montreal, Canada.

[19] Jacot, A., Gabriel, F., und Hongler, C. (2020). Neural tangent kernel: convergence and generalization in neural networks. *arXiv: 1806.07572v4.*

[20] Kawaguchi, K. (2016). Deep learning without poor local minima. *30th Conference on Neural Information Processing Systems (NIPS 2016).* Barcelona, Spain.

[21] Kawaguchi, K, and Huang, J. (2019). Gradient descent finds global minima for generalizable deep neural networks of practical sizes. *arXiv: 1908.02419v1.*

[22] 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* **25**, pp. 1097-1105. Red Hook, NY: Curran.

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

[24] Kohler, M., and Krzyżak, A. (2022). Regularized over-parametrized neural networks learned by gradient descent can generalize well. Submitted for publication.

[25] Kohler, M., Krzyżak, A., and Walter, B. (2022). On the rate of convergence of image classifiers based on convolutional neural networks. To appear in *Annals of the Institute of Statistical Mathematics.*

[26] Kohler, M., and Langer, S. (2020). Statistical theory for image classification using deep convolutional neural networks with cross-entropy loss. arXiv: 2011.13602.

[27] Kohler, M., and Walter, B. (2022). Analysis of convolutional neural network image classifiers in a rotationally symmetric model. Submitted for publication.

[28] 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* **25**, pp. 1097-1105. Red Hook, NY: Curran.

[29] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E.,Hubbard, W. and Jackel, L.D. (1989). Backpropagation applied to handwritten zip code recognition, *Neural Computation*, **1**, pp. 541-551.

[30] LeCun, Y., Bengio, Y. and Hinton, G. (2015). Deep learning. *Nature*, **521**, pp.436-444.

[31] Liang, S., Sun, R., Lee, J., and Srikant, R. (2018). Adding one neuron can eliminate all bad local minima. *Proceedings of the 32nd Conference on Neural Information Processing Systems (NIPS 2018)*, pp. 4355 - 4365. Montreal, Canada.

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

[33] Nitanda, A., and Suzuki, T. (2017). Stochastic particle gradient descent for infinite ensembles. *arXiv:1712.05438.*

[34] Rawat, W., and Wang, Z. (2017). Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Computation*, **29**, pp. 2352-2449.

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

[36] Walter, B. (2021). Analysis of convolutional neural network image classifiers in a hierarchical max-pooling model with additional local pooling. arXiv: 2106.05233

[37] Wang, H. and Lin, W. (2021). Harmless overparametrization in two-layer neural networks. *arXiv: 2106.04795v1.*

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

[39] Wyner, J. A., Olson, M., Bleich, J., and Mease, D. (2017) Explaining the success of AdaBost and random forest as interpolating classifiers. *The Journal of Machine Learning Research*, **18**, pp. 1558-1590.