Journal of Statistical Planning and Inference 234 (2025) 106188

Contents lists available at ScienceDirect

Journal of Statistical Planning and Inference

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

Statistical theory for image classification using deep convolutional neural network with cross-entropy loss under the hierarchical max-pooling model

Michael Kohler [a], Sophie Langer [b,*]

[a] Fachbereich Mathematik, Technische Universität Darmstadt, Schlossgartenstr. 7, 64289 Darmstadt, Germany
[b] Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, Drienerlolaan 5, 7522 NB Enschede, The Netherlands

ARTICLE INFO

Keywords:
Convolutional neural networks
Convergence rates
Curse of dimensionality
Image classification
Statistical risk bounds

ABSTRACT

Convolutional neural networks (CNNs) trained with cross-entropy loss have proven to be extremely successful in classifying images. In recent years, much work has been done to also improve the theoretical understanding of neural networks. Nevertheless, it seems limited when these networks are trained with cross-entropy loss, mainly because of the unboundedness of the target function. In this paper, we aim to fill this gap by analysing the rate of the excess risk of a CNN classifier trained by cross-entropy loss. Under suitable assumptions on the smoothness and structure of the a posteriori probability, it is shown that these classifiers achieve a rate of convergence which is independent of the dimension of the image. These rates are in line with the practical observations about CNNs.

1. Introduction

Deep convolutional neural networks (CNNs) have led to state-of-the-art performance in solving various problems, especially visual recognition tasks, see, e.g., LeCun et al. (2015), Krizhevsky et al. (2012), Schmidhuber (2015) and Rawat and Wang (2017). While deep learning applications are characterised above all by a high degree of flexibility, ranging from different initialisation strategies (Larochelle et al., 2009) to the choice of the right activation function (Janocha and Czarnecki, 2017) and the application of a proper learning algorithm (Le et al., 2011), one thing has so far been chosen as fixed for classification: The *log* or *cross entropy* loss. The smoothness of this loss function simplifies the optimisation procedure and shows good practical performance (Goodfellow et al., 2016; Simonyan and Zisserman, 2015). However, statistical risk bounds for neural networks trained with logistic loss only exist for very restrictive conditions (see, e.g., Kim et al. (2021)), mainly because of the unboundedness of the corresponding excess φ-risk minimiser (see (2)), which leads to slow convergence rates.

In general, many results on CNNs are based on considering them as a special type of feedfoward neural networks (FNNs) and then using results on FNNs to derive theoretical properties (Oono and Suzuki, 2019 and the literature cited therein). Unfortunately, these results do not demonstrate situations, where CNNs outperform FNNs, which is the case in many practical applications, especially in image classification. Generalisation bounds for CNNs with arbitrarily ordered fully connected and convolutional layers were derived in Lin and Zhang (2019). Here the model complexity is bounded by the norm of the convolutional weights leading to tighter bounds than existing bounds for FNNs. Yarotsky (2021) obtained approximation properties of deep CNNs, but only in an abstract setting, where it is unclear how to apply those results. Kohler et al. (2022) analysed plug-in classifiers based on CNNs and showed that

* Corresponding author.
E-mail addresses: kohler@mathematik.tu-darmstadt.de (M. Kohler), s.langer@utwente.nl (S. Langer).

https://doi.org/10.1016/j.jspi.2024.106188
Received 1 September 2023; Received in revised form 4 February 2024; Accepted 18 April 2024
Available online 5 June 2024

under proper assumptions on the structure of the a posteriori probability, suitable defined CNNs trained by squared loss achieve a rate of convergence which does not depend on the input dimension of the image. But as, e.g., experimental results in Golik et al. (2013) show, CNNs learned by cross entropy loss allow to find a better local optimum than the squared loss criterion. Therefore, CNNs learned by cross entropy loss are of higher practical relevance.

Cross-entropy loss or, more generally, convex surrogate loss functions have been studied in Bartlett et al. (2006) and Zhang (2004). Bartlett et al. (2006) showed that for convex loss functions ϕ satisfying a certain uniform strict convexity condition, the rate of convergence can be strictly faster than the classical $n^{-1/2}$, depending on the strictness of convexity of ϕ and the complexity of the class of classifiers. Zhang (2004) analysed how close the optimal Bayes error rate can be approximately reached using a classification algorithm that computes a classifier by minimising a convex upper bound of the classification error function. Some results of this article (see Lemma 1) are also used in our analysis. In Lemma 1(b) we derive a modification of Zhang's bound which enables us to derive better rate of convergence under proper assumptions on the a posteriori probability.

In this paper we derive dimension-free rates for CNN classifiers with cross-entropy loss in a binary image classification problem, where we impose some hierarchical structure on the a posteriori probability. In case that with high probability the a posteriori probability is very close to zero or one, meaning that the optimal classification rule makes only a very small error, our rate can even been improved. The first result can be framed as an extension of the analysis of Kohler et al. (2022), which analysed plug-in classifiers based on a class of CNNs in a similar setting. However, it is not straightforward to extend these results to CNNs with cross-entropy loss as one cannot analyse the classification problem as a nonparametric regression setting and a network approximation for the logistic function is needed. We deal with these difficulties with novel approximation results as well as an alternative proof strategy to bound the excess risk of the classifier.

2. Problem setting

2.1. Image classification

We consider a binary classification problem. Let $d_1, d_2 \in \mathbb{N}$, $\mathcal{X} = [0,1]^{d_1 \times d_2}$ be an image space and $\mathcal{Y} = \{-1, 1\}$ the set of corresponding binary labels. We describe an (random) image from a (random) class $Y \in \mathcal{Y}$ by a (random) matrix $\mathbf{X} \in \mathcal{X}$ with d_1 columns and d_2 rows, which contains at position (i, j) the grey scale value of the pixel of the image at the corresponding position. Let $\mathbf{P}$ be the probability measure of $\mathcal{X} \times \mathcal{Y}$ and define by

$$\eta(\mathbf{x}) = \mathbf{P}(Y = 1 | \mathbf{X} = \mathbf{x})$$

the so-called *a posteriori probability*.

Our aim is to predict Y by a deterministic function $g : \mathcal{X} \to \mathbb{R}$ such that the sign of $g(\mathbf{X})$ is a *good* prediction of Y. In particular, we aim to minimise the *prediction error* or 0-1 *risk*

$$\mathcal{E}(g) = \mathbf{P}(Y \operatorname{sgn}(g(\mathbf{X})) \leq 0) = \mathbf{E}(\mathbb{1}(\operatorname{sgn}(g(\mathbf{X})) \neq Y)),$$

where $\operatorname{sgn}(x) = 1$ if $x > 0$ and -1 otherwise and $\mathbb{1}(E)$ is the indicator function of the set E, that is, 1 if event E occurs and 0 otherwise. It is well-known, that the Bayes classifier $f^*(\mathbf{x}) = 2\eta(\mathbf{x}) - 1$ minimises $\mathcal{E}$ among all measurable functions (cf., e.g., Theorem 2.1 in Devroye et al. (1996)). But, as the probability measure $\mathbf{P}$ of $(\mathbf{X}, Y)$ is unknown in practice, we cannot find f^*. Instead we estimate f^* by using the training data $D_n = \{(\mathbf{X}_i, Y_i)\}_{i=1}^n$, where $(\mathbf{X}_i, Y_i)$ are independent copies of the random vector $(\mathbf{X}, Y) \sim \mathbf{P}$. A popular approach is estimating f^* by minimising the empirical risk

$$\mathcal{E}_n(g) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{\operatorname{sgn}(g(\mathbf{X}_i)) \neq Y_i\}$$

among a class of real-valued functions $\mathcal{F}_n$. However minimising the empirical risk with 0-1 loss over $\mathcal{F}_n$ is NP hard and thus computationally not feasible (Bartlett et al., 2006). By replacing the number of misclassifications by a convex surrogate loss φ, one can overcome computational problems. For a given loss φ we are searching for an estimate $\hat{f}_n \in \mathcal{F}_n$ that minimises

$$\mathcal{E}_n^{\varphi}(g) = \frac{1}{n} \sum_{i=1}^{n} \varphi\left(Y_i g(\mathbf{X}_i)\right).$$

By the law of large numbers, the empirical φ-risk converges to the population φ-risk

$$\mathcal{E}^{\varphi}(g) = \mathbf{E}(\varphi(Y g(\mathbf{X})))$$

when $n \to \infty$. A wide variety of classification methods are based on the idea of replacing the 0-1 risk by some kind of convex surrogate loss. In particular, AdaBoost (Friedman et al., 2000) employs the *exponential* loss $\exp(-z)$, while support vector machines often use *hinge* loss $\max(1-z, 0)$ (Vapnik, 1998) and logistic regression applies the *log* loss $\log(1+\exp(-x))$ (Hastie et al., 2009). In the context of CNNs and image classification it is a standard to use *cross-entropy* loss or *log* loss. Therefore we fix $\varphi(x) = \log(1+\exp(-x))$ in the following.

The classification performance of an estimator

$$\hat{f}_n \in \arg\min_{g \in \mathcal{F}_n} \mathcal{E}_n^{\varphi}(g) \tag{1}$$

is measured by its *excess risk*

$$\mathcal{E}(\hat{f}_n, f^*) = \mathcal{E}(\hat{f}_n) - \mathcal{E}(f^*).$$

Accordingly we denote the *excess φ-risk* by

$$\mathcal{E}^\varphi(\hat{f}_n, f_\varphi^*) = \mathcal{E}^\varphi(\hat{f}_n) - \mathcal{E}^\varphi(f_\varphi^*),$$

where

$$f_\varphi^* = \operatorname*{arg\,min}_{g \in \mathcal{F}_n} \mathcal{E}^\varphi(g) = \log\left(\frac{\eta(\mathbf{x})}{1-\eta(\mathbf{x})}\right) \tag{2}$$

in case of $\varphi(x) = \log(1+\exp(-x))$ (cf., Friedman et al. (2000)). Our following result states a relation between the excess risk and its logistic surrogate counterpart.

Lemma 1. *Define $\hat{f}_n$, f^* and f_φ^* as above.*

***(a)** Then*

$$\mathbf{E}\{\mathcal{E}(\hat{f}_n, f^*)\} \le \sqrt{2} \cdot \mathbf{E}\{\mathcal{E}^\varphi(\hat{f}_n, f_\varphi^*)^{1/2}\}$$

holds.

***(b)** Then*

$$\mathbf{E}\{\mathcal{E}(\hat{f}_n, f^*)\} \le 2 \cdot \mathbf{E}\{\mathcal{E}^\varphi(\hat{f}_n, f_\varphi^*)\} + 4 \cdot \mathcal{E}^\varphi(f_\varphi^*)$$

holds.

In both parts the expectation is taken over the training data $\mathcal{D}_n$.

Remark 1. We use two different bounds on the excess risk as in case of proper assumption on the distribution of $(\mathbf{X}, Y)$ (see Assumption 2 below), $\mathcal{E}^\varphi(f_\varphi^*)$ is small, such that part (b) of this lemma leads to faster rates.

Part (a) follows from Theorem 2.1 in Zhang (2004), where we choose $s = 2$ and $c = 2^{-1/2}$. For part (b) we set $\bar{f}_n(\mathbf{x}) := 1/(1+\exp(-\hat{f}_n(\mathbf{x})))$ and $g(z) := \log(z/(1-z))$ for $z \in (0,1)$. One can show that

$$\mathbf{E}\{\mathcal{E}(\hat{f}_n, f^*)\} \le 2 \cdot \mathbf{E}\{|\bar{f}_n(\mathbf{X}) - \eta(\mathbf{X})|\}. \tag{3}$$

For $h_1(z) := \varphi(1 \cdot g(z))$ and $h_2(z) := \varphi(-1 \cdot g(z))$ it further holds that $|h_j'(z)| \ge 1$ for $j \in \{1,2\}$ and $z \in (0,1)$. Using mean value theorem we can bound (3) by

$$2 \cdot \mathbf{E}\left\{\left|\varphi\left(Y \cdot g(\bar{f}_n(\mathbf{X}))\right) - \varphi\left(Y \cdot g(\eta(\mathbf{X}))\right)\right|\right\}.$$

With $|a-b| \le a+b$ for $a, b \ge 0$, the assertion follows. The complete proof is found in the appendix.

2.2. Hierarchical max-pooling model

In order to derive nontrivial rate of convergence results on the excess φ-risk of any estimate it is necessary to restrict the class of distributions (cf., Cover (1968) and Devroye (1982)). In case of logistic loss we have $f_\varphi^*(\mathbf{x}) = \log(\eta(\mathbf{x})/(1-\eta(\mathbf{x})))$, showing that f_φ^* is a monotone transformation of the a posteriori probability η. Hence we need to impose some assumptions on η.

As in Kohler et al. (2022) we assume that the a posteriori probability fulfils some *(p, C)-smooth hierarchical max-pooling model*. As smoothness measure we use the following definition of (p, C)-smoothness. For simplicity we introduce the multi-index notation, that is, $\partial^{\boldsymbol{\alpha}} = \partial^{\alpha_1} \dots \partial^{\alpha_d}$ with $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_d) \in \mathbb{N}_0^d$.

Definition 1. Let $p = q + s$ for some $q \in \mathbb{N}_0$ and $0 < s \le 1$. A function $f : \mathbb{R}^d \to \mathbb{R}$ is called (p, C)-**smooth**, if for every $\boldsymbol{\alpha} \in \mathbb{N}_0^d$ with $\|\boldsymbol{\alpha}\|_1 = q$ the partial derivative $\partial^{\boldsymbol{\alpha}} f$ exists and satisfies

$$|\partial^{\boldsymbol{\alpha}} f(\mathbf{x}) - \partial^{\boldsymbol{\alpha}} f(\mathbf{z})| \le C \cdot \|\mathbf{x} - \mathbf{z}\|^s$$

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

For the next definitions we frequently use the following notation: For $M \subseteq \mathbb{R}^d$ and $\mathbf{x} \in \mathbb{R}^d$ we define

$$\mathbf{x} + M = \{\mathbf{x} + \mathbf{z} \ : \ \mathbf{z} \in M\}.$$

For $I \subseteq d_1 \times d_2$ and $\mathbf{x} = (x_i)_{i \in d_1 \times d_2} \in [0,1]^{d_1 \times d_2}$ we set

$$\mathbf{x}_I = (x_i)_{i \in I}.$$

The definition of *hierarchical max-pooling models* is motivated by the following observation: Human beings often decide, whether a given image contains some object, i.e., a car, or not by scanning subparts of the image and checking, whether the searched object

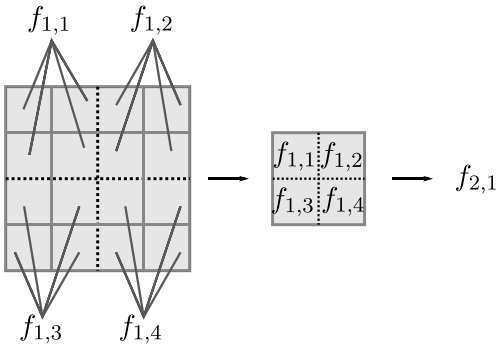

Fig. 1. Illustration of Definition 3 with $l = 2$.

is on this subpart. For each subpart the human estimates a probability that the searched object is on it. The probability that the whole image contains the object is then simply the maximum of the probabilities for each subpart of the image. This leads to the definition of a max-pooling model for the a posteriori probability.

Definition 2. Let $d_1, d_2 \in \mathbb{N}$ with $d_1, d_2 > 1$ and $m : [0,1]^{d_1 \times d_2} \to \mathbb{R}$. We say that m satisfies a **max-pooling model with index set**

$$I \subseteq d_1 - 1 \times d_2 - 1,$$

if there exist a function $f : [0,1]^{(1,1)+I} \to \mathbb{R}$ such that

$$m(\mathbf{x}) = \max_{(i,j) \in \mathbb{Z}^2 \,:\, (i,j)+I \subseteq d_1 \times d_2} f\left(\mathbf{x}_{(i,j)+I}\right)$$

for $\mathbf{x} \in [0,1]^{d_1 \times d_2}$.

Additionally, the probability that a subpart contains the searched object is composed by several decisions, if parts of the searched objects are identifiable. This motivates the hierarchical structure of our model. In the following we denote the four block matrices of a matrix $\mathbf{x} \in [0,1]^{2^k \times 2^k}$ by $\mathbf{x}_{1,1}, \mathbf{x}_{2,1}, \mathbf{x}_{1,2}, \mathbf{x}_{2,2}$, where

$$\mathbf{x}_{i,j} = \mathbf{x}_{\{(i-1)2^{k-1}+1,\dots,i2^{k-1}\} \times \{(j-1)2^{k-1}+1,\dots,j2^{k-1}\}} \in [0,1]^{2^{k-1} \times 2^{k-1}},$$

$i, j \in \{1,2\}$. This means that

$$\mathbf{x} = \left(\begin{array}{c|c} \mathbf{x}_{1,1} & \mathbf{x}_{2,1} \\ \hline \mathbf{x}_{1,2} & \mathbf{x}_{2,2} \end{array}\right). \tag{4}$$

Definition 3. Let $d_1, d_2 \in \mathbb{N}$ with $d_1, d_2 > 1$ and $m : [0,1]^{d_1 \times d_2} \to \mathbb{R}$. We say that

$$f : [0,1]^{2^l \times 2^l} \to \mathbb{R}$$

satisfies a **hierarchical model of level** l, if there exist functions

$$g_{k,s} : \mathbb{R}^4 \to [0,1] \quad (k = 1, \dots, l, s = 1, \dots, 4^{l-k})$$

such that we have

$$f = f_{l,1}$$

for some $f_{k,s} : [0,1]^{2^k \times 2^k} \to \mathbb{R}$ recursively defined by

$$f_{k,s}(\mathbf{x}) = g_{k,s}\left(f_{k-1,4\cdot(s-1)+1}(\mathbf{x}_{1,1}), f_{k-1,4\cdot(s-1)+2}(\mathbf{x}_{2,1}),\ f_{k-1,4\cdot(s-1)+3}(\mathbf{x}_{1,2}), f_{k-1,4\cdot s}(\mathbf{x}_{2,2})\right)$$

for $k = 2, \dots, l, s = 1, \dots, 4^{l-k}$ and $\mathbf{x} \in [0,1]^{2^k \times 2^k}$ and

$$f_{1,s}(x_{1,1}, x_{1,2}, x_{2,1}, x_{2,2}) = g_{1,s}(x_{1,1}, x_{1,2}, x_{2,1}, x_{2,2})$$

for $s = 1, \dots, 4^{l-1}$ and $x_{1,1}, x_{1,2}, x_{2,1}, x_{2,2} \in [0,1]$.

An illustration of Definition 3 for $l = 2$ is shown in Fig. 1.

Combining Definitions 1–3 leads to the final definition of *(p, C)-smooth hierarchical max-pooling models*.

Definition 4. We say that $m : [0,1]^{d_1 \times d_2} \to \mathbb{R}$ satisfies a (p, C)**-smooth hierarchical max-pooling model of level** l (where $2^l \leq \min\{d_1, d_2\}$), if m satisfies a max-pooling model with index set $I = 2^l - 1 \times 2^l - 1$, the function $f : [0,1]^{(1,1)+I} \to \mathbb{R}$ in the definition of this max-pooling model satisfies a hierarchical model with level l and if all functions $g_{k,s}$ in the definition of the functions m are (p, C)–smooth for some $C > 0$.

2.3. Convolutional neural networks

We consider CNNs that take $d_1 \times d_2$-dimensional images as input and produce an one-dimensional output. As the name suggests, the most important operation of a CNN is its convolution. The main idea behind it is to apply filters, i.e., small weight matrices to the input image to extract high-level information. Mathematically a convolution can be described as follows: Let $\mathbf{X}$ be a $d_1 \times d_2$ input matrix, $\mathbf{X}_{i,j}$ be its $\ell \times \ell$ block matrix with entries $(X_{i+a,j+b})_{a,b=0,\dots,\ell-1}$ and $\mathbf{W}$ be a corresponding filter of size ℓ. The entry (i,j) of a resulting channel $\tilde{\mathbf{C}}$ can then be described by

$$\tilde{C}_{i,j} = \sum_{k,m=1}^{\ell} (\mathbf{X}_{i,j} \odot \mathbf{W})_{k,m}, \tag{5}$$

where $\odot$ denotes the Hadamard product. Finally an activation function σ is applied componentwise, i.e., $C_{i,j} := \sigma(\tilde{C}_{i,j})$. This in turn means that the final channel $\mathbf{C}$ consists of entries computed by the sum of a Hadamard product between the filter and the respective block matrix of the input applied to an activation function σ. We set

$$\mathbf{C} := \sigma(\mathbf{W} \star \mathbf{X})$$

with $\sigma(x) = \max\{x, 0\}$ being the ReLU activation function. Here $\star$ describes the computation of each entry $\tilde{C}_{i,j}$ as in (5), where σ is applied componentwise. One can see that the weights generating the feature map $\mathbf{C}$ are shared, which has the advantage of reducing the complexity of the model and the training time of the networks. Usually a CNN consists of several convolutional layers. Each convolutional layer l ($l \in \{1, \dots, L\}$) consists of $k_l \in \mathbb{N}$ channels (also called feature maps) while the filter size $M_l \in \{1, \dots, \min\{d_1, d_2\}\}$ per layer is fixed. In our setting we make use of so-called zero-padding meaning that we enlarge each channel by appending zero matrices on each side such that the convolution does not change the in-plane dimension. This, in turn, means that every resulting channel has size $d_1 \times d_2$. For filters

$$(\mathbf{W}_{s_1,s_2,l})_{l=1,\dots,L,s_1=1,\dots,k_{l-1},s_2=1,\dots,k_l}$$

the s-th channel of layer l ($s = 1, \dots, k_l, l = 1, \dots, L$) can be described by

$$\mathbf{C}_{s,l} = \sigma\left(\sum_{s_1=1}^{k_{l-1}} \mathbf{W}_{s_1,s,l} \star \mathbf{C}_{s_1,l-1}\right) \tag{6}$$

with $\mathbf{C}_{1,0} = \mathbf{X}$ and $k_0 = 1$.

In our network, only in the last step a max-pooling layer is applied to the values of the last convolutional layer L. As in Langer and Schmidt-Hieber (2022) we consider a global max-pooling where we extract from every channel $\mathbf{C}_{s,L}$ ($s = 1, \dots, k_L$) the largest absolute value. A CNN with $L \in \mathbb{N}$ convolutional layers and one pooling layer, channel vector $\mathbf{k} = (k_1, \dots, k_L) \in \mathbb{N}^L$ and filter vector $\mathbf{M} = (M_1, \dots, M_L) \in \mathbb{N}^L$, where k_i describes the number of channels and M_i describes the size of the filters in layer i, respectively, can be described as a function $f : [0,1]^{d_1 \times d_2} \to \mathbb{R}^{k_L}$ with

$$\mathbf{x} \mapsto f(\mathbf{x}) = (|\mathbf{C}_{1,L}|_\infty, \dots, |\mathbf{C}_{k_L,L}|_\infty)$$

with $\mathbf{C}_{s,L}$ recursively defined as in (6). We denote this network class by $\mathcal{F}^C_{L,\mathbf{k},\mathbf{M}}$.

After convolutional and pooling layers typically several fully connected layers are applied. Again we choose the ReLU activation function $\sigma(x) = \max\{x, 0\}$. Following the definition in Schmidt-Hieber (2020), a fully connected network with $L \in \mathbb{N}$ hidden layers and width vector $\mathbf{k} = (k_0, \dots, k_{L+1}) \in \mathbb{N}^{L+2}$, where k_i denotes the number of neurons in layer i, can be described by a function $f : \mathbb{R}^{k_0} \to \mathbb{R}^{k_{L+1}}$ with

$$\mathbf{x} \mapsto f(\mathbf{x}) = \mathbf{W}_L \sigma_{v_L} \mathbf{W}_{L-1} \sigma_{v_{L-1}} \cdots \mathbf{W}_1 \sigma_{\mathbf{v}_1} \mathbf{W}_0 \mathbf{x},$$

where $\mathbf{W}_j$ is a $k_j \times k_{j+1}$ weight matrix and $\mathbf{v}_j$ is the j-th shift vector. We denote the network class of fully connected neural networks by $\mathcal{F}_{L,\mathbf{k}}$.

Our final function class $\mathcal{F}_{n,\Theta}$ is then a composition of convolutional and fully connected layers, i.e.,

$$\mathcal{F}_{n,\Theta} = \left\{ g \circ f : f \in \mathcal{F}^C_{L_n^{(1)},\mathbf{k}^{(1)},\mathbf{M}}, g \in \mathcal{F}_{L_n^{(2)},\mathbf{k}^{(2)}}, \|g \circ f\|_\infty \le \beta_n \right\}, \tag{7}$$

where $\Theta = (\mathbf{L}, \mathbf{k}^{(1)}, \mathbf{k}^{(2)}, \mathbf{M})$ with parameters

$$\mathbf{L} = (L_n^{(1)}, L_n^{(2)}),\ \mathbf{k}^{(1)} = \left(k_1^{(1)}, \dots, k^{(1)}_{L_n^{(1)}}\right),$$
$$\mathbf{k}^{(2)} = \left(k_1^{(2)}, \dots, k^{(2)}_{L_n^{(2)}}\right),\ \mathbf{M} = (M_1, \dots, M_{L_n^{(1)}})$$

and $\beta_n = c_1 \cdot \log n$ for some constant $c_1 > 0$. Accordingly we denote by

$$\hat{f}_n^{CNN} = \arg\min_{f \in \mathcal{F}_{n,\Theta}} \frac{1}{n} \sum_{i=1}^{n} \log(1 + \exp(-Y_i \cdot f(\mathbf{X}_i))) \tag{8}$$

the empirical risk minimiser based on our class of CNNs.

In general, deep learning theory can be roughly divided into three parts, namely expressivity, generalisation and optimisation (see Kutyniok (2020)). While the intersection of all three aspects has only been analysed in very limited settings so far, e.g., for shallow neural networks and a restricted class of regression functions (see, e.g., Braun et al. (2021)), most statistical risk bounds of neural networks exclude the optimisation algorithm and deal with the empirical risk minimiser (ERM) instead (see, e.g., Schmidt-Hieber (2020), Bauer and Kohler (2019) and Kohler and Langer (2021)). Following this line of work, we also analyse the ERM based on a particular class of CNNs. It therefore remains an open question whether similar rates can be shown for CNNs trained with (stochastic) gradient descent. In case of overparametrized CNNs, e.g., Du et al. (2019) could show that the gradient descent is able to find the global minimum of the empirical loss function. But for overparametrized networks one cannot use standard generalisation bounds as these usually depend on the number of parameters. Therefore a completely new statistical approach is needed for this analysis.

3. Main result

In this section we derive convergence rates of the excess risk of $\hat{f}_n^{CNN}$ under the assumption that the a posteriori probability η fulfils a (p, C)-smooth hierarchical max-pooling model (see Definition 4). Before providing our results, two assumptions are imposed on the distribution of $(\mathbf{X}, Y)$.

Assumption 1. For $p \geq 1$ and $C > 0$ arbitrary, $\eta(\mathbf{x}) = \mathbf{P}\{Y = 1 | \mathbf{X} = \mathbf{x}\}$ satisfies a (p, C)–smooth hierarchical max-pooling model of finite level l and $supp(\mathbf{P}_{\mathbf{X}}) \subseteq [0,1]^{d_1 \times d_2}$.

The second is a margin condition on the a posteriori probability.

Assumption 2. For f_φ^* being the minimiser of $\mathcal{E}^\varphi(g)$ and $n \in \mathbb{N}$, it holds

$$\mathbf{P}\left\{\mathbf{X} : |f_\varphi^*(\mathbf{X})| > \frac{1}{2} \cdot \log n\right\} \geq 1 - \frac{1}{\sqrt{n}}.$$

Assumption 2 requires that with high probability the a posteriori probability is very close to zero or one, and hence the optimal classification rule makes only a very small error. This is in particular realistic for various image classification tasks, where object classes can often be confidently distinguished (cf., Kim et al. (2021)). Additionally, a similar assumption was applied in Bos and Schmidt-Hieber (2022) within the context of a multiclass classification problem, also analysing a ReLU network classifier minimising cross entropy loss.

Theorem 1. *Suppose Assumption 1 holds. Set*

$$L_n^{(1)} = \frac{4^l - 1}{3} \cdot \lceil c_3 \cdot n^{2/(2p+4)} \rceil + l \quad \text{and} \quad L_n^{(2)} = \lceil c_2 \cdot n^{1/4} \rceil,$$

$$M_s = 2^{\pi(s)} \quad (s = 1, \dots, L_n^{(1)}),$$

where the function $\pi : \{1, \dots, L_n^{(1)}\} \to \{1, \dots, l\}$ *is defined by*

$$\pi(s) = \sum_{i=1}^{l} \mathbb{1}_{\{s \geq i + \sum_{r=l-i+1}^{l-1} 4^r \cdot \lceil c_3 \cdot n^{2/(2p+4)} \rceil\}},$$

choose $\mathbf{k}^{(1)} = (c_4, \dots, c_4) \in \mathbb{N}^{L_n^{(1)}}$ *and* $\mathbf{k}^{(2)} = (c_5, \dots, c_5) \in \mathbb{N}^{L_n^{(2)}}$, *and define the estimate* $\hat{f}_n^{CNN}$ *as in (8). Assume that the constants* $c_2, \dots, c_5$ *are sufficiently large.*

(a) *There exists a constant* $c_6 = c_6(\eta, d_1, d_2, p, C, l) > 0$ *such that we have for any* $n > 1$

$$\mathbf{E}\{\mathcal{E}(\hat{f}_n^{CNN}, f^*)\} \leq c_6 \cdot (\log n) \cdot n^{-\min\{\frac{p}{4p+8}, \frac{1}{8}\}}.$$

(b) *If, in addition, Assumption 2 holds , then there exists a constant* $c_7 = c_7(\eta, d_1, d_2, p, C, l) > 0$ *such that we have for any* $n > 1$

$$\mathbf{E}\{\mathcal{E}(\hat{f}_n^{CNN}, f^*)\} \leq c_7 \cdot (\log n)^2 \cdot n^{-\min\{\frac{p}{2p+4}, \frac{1}{4}\}}.$$

In both parts the expectation is taken over the training data $\mathcal{D}_n$.

Remark 2. An interesting feature of the convergence rates in Theorem 1 is that both rates do not depend on the dimension $d_1 \cdot d_2$ of the input image. Thus, given the structure of the a posteriori probability fulfils a (p, C)-smooth hierarchical max-pooling model, our estimator circumvents the curse of dimensionality. Under Assumption 2 the rate can even be improved. To us these results partly explain the good performance of CNN classifiers on image data.

Remark 3. The definition of the parameters $L_n^{(1)}$ and M_i $(i = 1, \dots, L_n^{(1)})$ of the estimate in Theorem 1 depends on the smoothness and the level of the hierarchical max-pooling model for the a posteriori probability, which are usually unknown in applications. In this case it is possible to define these parameters in a data-dependent way, e.g., by using a splitting of the sample approach (cf., e.g., Chapter 7 in Györfi et al. (2002)).

On the proof. To prove Theorem 1 we use Lemma 1 in combination with the following general upper bound on the excess φ-risk of an empirical risk minimiser $\hat{f}_n \in \mathcal{F}_n$, where $\mathcal{F}_n$ can be a general function space consisting of functions $f : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}$.

Lemma 2. *Let φ be the logistic loss and $\mathcal{D}_n = \{(\mathbf{X}_i, Y_i)\}_{i=1}^n$. Then the empirical risk minimiser $\hat{f}_n$ defined as in (1) satisfies*

$$\mathbf{E}\{\mathcal{E}^\varphi(\hat{f}_n, f_\varphi^*)\} \le 2 \cdot \sup_{f \in \mathcal{F}_n} |\mathcal{E}^\varphi(f) - \mathcal{E}_n^\varphi(f)| + \inf_{f \in \mathcal{F}_n} \mathcal{E}^\varphi(f, f_\varphi^*).$$

Lemma 2 shows that the excess φ-risk of an ERM is bounded above by the sum of two terms. The first term is the so-called *generalisation* error. It is closely related to the complexity of the function class and can be bounded using results from empirical process theory. The second one is the *approximation error* measuring how rich the function class $\mathcal{F}_n$ is, meaning if we can express the problem under consideration by a function of $\mathcal{F}_n$.

In the following $\mathcal{F}_n$ is chosen to be a class of convolutional neural networks, i.e., $\mathcal{F}_n = \mathcal{F}_{n,\Theta}$ and the estimator under consideration is defined as in (8).

4. Approximation error

To bound $\inf_{f \in \mathcal{F}_{n,\Theta}} \mathcal{E}^\varphi(f, f_\varphi^*)$ we use that for an arbitrary $h \in \mathcal{F}_{n,\Theta}$

$$\begin{aligned}
\inf_{f \in \mathcal{F}_{n,\Theta}} \mathcal{E}^\varphi(f, f_\varphi^*) \le \mathcal{E}^\varphi(h, f_\varphi^*) &= \int (\eta(\mathbf{x}) \cdot \varphi(h(\mathbf{x})) + (1 - \eta(\mathbf{x})) \cdot \varphi(-h(\mathbf{x})))\mathbf{P}_\mathbf{X}(d\mathbf{x}) \\
&\quad - \int (\eta(\mathbf{x}) \cdot \varphi(f_\varphi^*(\mathbf{x})) + (1 - \eta(\mathbf{x})) \cdot \varphi(-f_\varphi^*(\mathbf{x})))\mathbf{P}_\mathbf{X}(d\mathbf{x}) \\
&\le \sup_{\mathbf{x} \in [0,1]^{d_1 \times d_2}} \Big(\Big| \eta(\mathbf{x}) \cdot \Big(\varphi(h(\mathbf{x})) - \varphi(f_\varphi^*(\mathbf{x})) \Big) \Big| \\
&\quad + \Big| (1 - \eta(\mathbf{x})) \cdot \Big(\varphi(-h(\mathbf{x})) - \varphi(-f_\varphi^*(\mathbf{x})) \Big) \Big| \Big) \\
&\le \sup_{\mathbf{x} \in [0,1]^{d_1 \times d_2}} \Big(|\eta(\mathbf{x}) \cdot (\varphi(h(\mathbf{x})) - \varphi(g(\eta(\mathbf{x}))))| \\
&\quad + |(1 - \eta(\mathbf{x})) \cdot (\varphi(-h(\mathbf{x})) - \varphi(-g(\eta(\mathbf{x}))))| \Big),
\end{aligned}$$

where

$$g(z) = \begin{cases} \infty & , z = 1 \\ \log \frac{z}{1-z} & , 0 < z < 1 \\ -\infty & , z = 0. \end{cases}$$

This, in turn, means that in order to find a satisfying bound for our approximation error we need to build a CNN which approximates $g(\eta(\mathbf{x}))$ properly. Using the compositional structure of neural networks, one can break this task down into two parts. On the one hand we show that CNNs approximate $\eta(\mathbf{x})$, i.e., (p, C)-smooth hierarchical max-pooling models. On the other hand we build a fully connected neural network that approximates g. The approximation result on g is the following.

Lemma 3. *Set*

$$g(z) = \begin{cases} \infty & , z = 1 \\ \log \frac{z}{1-z} & , 0 < z < 1 \\ -\infty & , z = 0 \end{cases}$$

and let $K \in \mathbb{N}$ with $K \ge 6$. Let $\eta : \mathbb{R}^d \to [0,1]$ and let $\bar{\eta} : \mathbb{R}^d \to \mathbb{R}$ such that $\|\bar{\eta} - \eta\|_\infty \le \epsilon$ for some $0 \le \epsilon \le 1/K$. Then there exists a neural network $\bar{g} : \mathbb{R} \to \mathbb{R}$ with ReLU activation function, $K+3$ hidden layers with 7 neurons per layer, which is bounded in absolute value by $\log(K+1)$ and which satisfies

$$\begin{aligned}
&\sup_{\mathbf{x} \in \mathbb{R}^{d_1 \times d_2}} (|\eta(\mathbf{x}) \cdot (\varphi(\bar{g}(\bar{\eta}(\mathbf{x}))) - \varphi(g(\eta(\mathbf{x}))))| \\
&\quad + |(1 - \eta(\mathbf{x})) \cdot (\varphi(-\bar{g}(\bar{\eta}(\mathbf{x}))) - \varphi(-g(\eta(\mathbf{x}))))|) \\
&\le c_{10} \cdot \left(\frac{\log K}{K} + \epsilon \right).
\end{aligned}$$

A complete proof is found in the appendix. Roughly, it is based on the idea that functions of the form

$$\bar{g}(z) := \sum_{k=-1}^{K+1} a_k \cdot B_k(z),$$

where

$$B_k(z) = \begin{cases} 0 & , z < \frac{k-1}{K} \\ K \cdot (z - \frac{k-1}{K}) & , \frac{k-1}{K} \le z < \frac{k}{K} \\ K \cdot (\frac{k+1}{K} - z) & , \frac{k}{K} \le z < \frac{k+1}{K} \\ 0 & , z \ge \frac{k+1}{K}, \end{cases}$$

can be computed by a ReLU network with $K+3$ hidden layers and 7 neurons per layer.

Combining Lemma 3 with the approximation result on the hierarchical max-pooling models, we are then able to show the following approximation result.

Theorem 2. *Suppose Assumption 1 holds. Let $\mathcal{F}_{n,\Theta}$ be the set of all CNNs with $\Theta = (\mathbf{L}, \mathbf{k}^{(1)}, \mathbf{k}^{(2)}, \mathbf{M})$, where $\mathbf{k}^{(1)} = (c_4, \dots, c_4)$ with c_4 sufficiently large and $\mathbf{k}^{(2)} = (7, \dots, 7)$.*

Furthermore assume $(L_n^{(1)})^{2p/d} \ge c_8 \cdot L_n^{(2)}$. Then

$$\inf_{f \in \mathcal{F}_{n,\Theta}} \mathcal{E}(f, f_\varphi^*) \le c_9 \cdot \left(\frac{\log L_n^{(2)}}{L_n^{(2)}} + \frac{1}{(L_n^{(1)})^{2p/4}} \right),$$

with constant $c_9 = c_9(\eta, p, C, l)$.

The complete proof of this result is given in the appendix.

5. Generalisation error

The generalisation error $\sup_{f \in \mathcal{F}_n} |\mathcal{E}^\varphi(f) - \mathcal{E}_n^\varphi(f)|$ can be bounded using results from empirical process theory together with bounds on the covering number of CNNs.

In particular, using Theorem 9.1 in Györfi et al. (2002) it holds for

$$\mathbf{Z} = (\mathbf{X}, Y), \mathbf{Z}_1 = (\mathbf{X}_1, Y_1), \dots, \mathbf{Z}_n = (\mathbf{X}_n, Y_n),$$

and $\epsilon > 0$, that

$$\begin{aligned} \mathbf{P}\left\{ \sup_{f \in \mathcal{F}_n} \left| \mathcal{E}^\varphi(f) - \mathcal{E}_n^\varphi(f) \right| > \epsilon \right\} &= \mathbf{P}\left\{ \sup_{h \in \mathcal{H}_n} \left| \mathbf{E}h(\mathbf{Z}) - \frac{1}{n} \sum_{i=1}^n h(\mathbf{Z}_i) \right| > \epsilon \right\} \\ &\le 8\mathbf{E}\left\{ \mathcal{N}_1\left(\frac{\epsilon}{8}, \mathcal{H}_n, \mathbf{Z}_1^n \right) \right\} e^{-\frac{n\epsilon^2}{128 \cdot c_{10}^2 \cdot (\log n)^2}}. \end{aligned}$$

Here

$$\mathcal{H}_n = \{ h : \mathbb{R}^{d_1 \times d_2} \times \mathbb{R} \to \mathbb{R} : \exists f \in \mathcal{F}_{n,\Theta} \text{ such that } h(\mathbf{x}, y) = \varphi(y \cdot f(\mathbf{x})) \}$$

and $\mathcal{N}_1(\epsilon, \mathcal{F}, \mathbf{x}_1^n)$ describes the ϵ–covering number of $\mathcal{F}$ on $\mathbf{x}_1^n$, that is the smallest ϵ–cover of $\mathcal{F}$ on $\mathbf{x}_1^n$, i.e., the number $N \in \mathbb{N}$ such that there exists $i \in \{1, \dots, N\}$ such that

$$\frac{1}{n} \sum_{k=1}^n |f(\mathbf{x}_k) - f_i(\mathbf{x}_k)| < \epsilon.$$

As every ϵ-cover of $\mathcal{F}_{n,\Theta}$ is an ϵ-cover of $\mathcal{H}_n$, we have

$$\mathcal{N}_1\left(\frac{\epsilon}{8}, \mathcal{H}_n, \mathbf{Z}_1^n \right) \le \mathcal{N}_1\left(\frac{\epsilon}{8}, \mathcal{F}_{n,\Theta}, \mathbf{X}_1^n \right).$$

The following bound on the covering number of $\mathcal{F}_{n,\Theta}$, then helps us to bound the generalisation error.

Lemma 4. *Define $\mathcal{F}_{n,\Theta}$ as in (7) and set*

$$k_{max} = \max\left\{ k_1^{(1)}, \dots, k_{L_n^{(1)}}^{(1)}, k_1^{(2)}, \dots, k_{L_n^{(2)}}^{(2)} \right\},$$

$$M_{max} = \max\{ M_1, \dots, M_{L_n^{(1)}} \}$$

and

$$L_{max} = \max\{ L_n^{(1)}, L_n^{(2)} \}.$$

Assume $d_1 \cdot d_2 > 1$ and $\beta_n = c_1 \cdot \log n \ge 2$. Then we have for any $\epsilon \in (0, 1)$,

$$\sup_{\mathbf{x}_1^n \in (\mathbb{R}^{d_1 \times d_2})^n} \log\left(\mathcal{N}_1\left(\epsilon, \mathcal{F}_{n,\Theta}, \mathbf{x}_1^n \right) \right)$$

$$\leq c_{11} \cdot L_{max}^2 \cdot \log(L_{max} \cdot d_1 \cdot d_2) \cdot \log\left(\frac{c_1 \cdot \log n}{\epsilon}\right)$$

for some constant $c_{11} > 0$ *which depends only on* k_{max} *and* M_{max}.

The proof of this result follows by Lemma 7 in Kohler et al. (2022).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.jspi.2024.106188.

References

Bartlett, P.L., Jordan, M.I., Mcauliffe, J.D., 2006. Convexity, classification, and risk bounds. J. Amer. Statist. Assoc. 101 (473), 138–156.

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

Bos, T., Schmidt-Hieber, J., 2022. Convergence rates of deep ReLU networks for multiclass classification. Electron. J. Stat. 16 (1), 2724–2773.

Braun, A., Kohler, M., Langer, S., Walk, H., 2021. The smoking gun: Statistical theory improves neural network estimates. arXiv preprint, arXiv:2107.09550.

Cover, T.M., 1968. Rates of convergence of nearest neighbor procedures. In: HICSS. pp. 413–415.

Devroye, L., 1982. Necessary and sufficient conditions for the pointwise convergence of nearest neighbor regression function estimates. Probab. Theory Related Fields 61 (4), 467–481.

Devroye, L., Györfi, L., Lugosi, G., 1996. A Probabilistic Theory of Pattern Recognition. Springer, New York.

Du, S., Lee, J., Li, H., Wang, L., Zhai, X., 2019. Gradient descent finds global minima of deep neural networks. In: ICLR. pp. 1675–1685.

Friedman, J., Hastie, T., Tibshirani, R., 2000. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). Ann. Statist. 28 (2), 337–407.

Golik, P., Doetsch, P., Ney, H., 2013. Cross-entropy vs. squared error training: a theoretical and experimental comparison. In: INTERSPEECH. pp. 1756–1760.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

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

Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning. Springer New York.

Janocha, K., Czarnecki, W., 2017. On loss functions for deep neural networks in classification. arXiv preprint, arxiv:1702.05659.

Kim, Y., Ohn, I., Kim, D., 2021. Fast convergence rates of deep neural networks for classification. Neural Netw. 138, 179–197.

Kohler, M., Krzyżak, A., Walter, B., 2022. On the rate of convergence of image classifiers based on convolutional neural networks. Ann. Inst. Stat. Math..

Kohler, M., Langer, S., 2021. On the rate of convergence of fully connected deep neural network regression estimates. Ann. Statist. 49 (4), 2231–2249.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. ImageNet classification with deep convolutional neural networks. In: NeurIPS. pp. 1097–1105, URL http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

Kutyniok, G., 2020. Discussion of "nonparametric regression using deep neural networks with ReLU activation function". Ann. Statist. 48 (4), 1902–1905.

Langer, S., Schmidt-Hieber, J., 2022. A statistical analysis of an image classification problem. arXiv preprint, arxiv:2206.02151.

Larochelle, H., Bengio, Y., Louradour, J., Lamblin, P., 2009. Exploring strategies for training deep neural networks. J. Mach. Learn. Res. 10 (1), 1–40, URL http://jmlr.org/papers/v10/larochelle09a.html.

Le, Q.V., Ngiam, J., Coates, A., Lahiri, A., Prochnow, B., Ng, A.Y., 2011. On optimization methods for deep learning. In: Proceedings of the 28th International Conference on International Conference on Machine Learning. ICML '11, Omni Press, Madison, WI, USA, pp. 265–272.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521 (7553), 436–444.

Lin, S., Zhang, J., 2019. Generalization bounds for convolutional neural networks. http://dx.doi.org/10.48550/ARXIV.1910.01487, arXiv preprint, arXiv:1910.01487.

Oono, K., Suzuki, T., 2019. Approximation and non-parametric estimation of ResNet-type convolutional neural networks via block-sparse fully-connected neural networks. In: Proceedings of the 36th International Conference on Machine Learning. URL https://openreview.net/forum?id=HklnzhR9YQ.

Rawat, W., Wang, Z., 2017. Deep convolutional neural networks for image classification: A comprehensive review. Neural Comput. 29 (9), 2352–2449.

Schmidhuber, J., 2015. Deep learning in neural networks: An overview. Neural Netw. 61, 85–117.

Schmidt-Hieber, J., 2020. Nonparametric regression using deep neural networks with ReLU activation function. Ann. Statist. 48 (4), 1875–1897.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. In: ICLR.

Vapnik, V.N., 1998. Statistical Learning Theory. Wiley-Interscience.

Yarotsky, D., 2021. Universal approximations of invariant maps by neural networks. Constr. Approx. 55 (1), 407–474.

Zhang, T., 2004. Statistical behavior and consistency of classification methods based on convex risk minimization. Ann. Statist. 32 (1), 56–85, URL http://www.jstor.org/stable/3448494.