Do highly over-parameterized neural networks generalize since bad solutions are rare?

Julius Martinetz and Thomas Martinetz, *Senior Member, IEEE*

***Abstract*—We study over-parameterized classifiers where Empirical Risk Minimization (ERM) for learning leads to zero training error. In these over-parameterized settings there are many global minima with zero training error, some of which generalize better than others. We show that under certain conditions the fraction of "bad" global minima with a true error larger than ε decays to zero exponentially fast with the number of training data n. The bound depends on the distribution of the true error over the set of classifier functions used for the given classification problem, and does not necessarily depend on the size or complexity (e.g. the number of parameters) of the classifier function set. This insight provides an alternative perspective on the unexpectedly good generalization even of highly over-parameterized neural networks. We substantiate our theoretical findings through experiments on synthetic data and a subset of MNIST. Additionally, we assess our hypothesis using VGG19 and ResNet18 on a subset of Caltech101.**

***Index Terms*—Over-parameterized classifier, generalization, small training data sets, Empirical Risk Minimization, zero training loss.**

I. Introduction

Extreme upscaling and deepening of neural networks has led to a quantum leap in performance in many real-world object recognition tasks [1]. The same is true for neural networks in Natural Language Processing [2] and reinforcement learning [3]. By enlarging the networks appropriately, the level of performance increases [4], and even entering regimes with many more network parameters than training samples does not necessarily harm generalization [5]. This is usually attributed to an appropriate regularization [6]. However, generalization is also observed in highly over-parameterized regimes and without explicit regularization, where even random label assignments or even random image data can be memorized [7]. Belkin called this the "modern" interpolation regime' [8], although this effect has been studied for some time [9].

The generalization capabilities of large neural networks do not seem to be harmed by the millions of parameters in today's popular architectures. Due to their large, inherent capacity the training error is zero after training by Empirical Risk Minimization (ERM), i.e. the training data is memorized. Nevertheless, the test error can be unexpectedly low [7], even after training on only a few handful of training samples and without explicit regularization [10]. This is in contrast to traditional machine learning wisdom, where one would not expect any generalization in these highly over-parameterized regimes, where the data is highly over-fitted. Statistical learning theory based on uniform generalization bounds requires a high probability that there is no solution with a large deviation between empirical and expected risk, i.e. the training and the true error. However, this is not the case in over-parameterized regimes. There is no uniform generalization bound. In these highly over-parameterized regimes always also "bad" solutions with zero training but large true error do exist. Interestingly, they hardly seem to occur in practice [7].

In this paper, we investigate over-parameterized classifiers with training error zero solutions. A classifier receives inputs x from an input distribution $P(x)$ and assigns a class label using a classifier function $h(x)$. The input x has a true label y with probability $P(y|x)$. For each input x, the classifier produces a loss $L(y, h(x)) \in \{0, 1\}$, 0 if the classification is correct and 1 if the classification is incorrect. Note, that this loss applies to binary- as well as multi-class scenarios. The error of the classifier h on the whole data distribution $P(x, y) = P(y|x)P(x)$ is given by the expected risk

$$E(h) = \int L(y, h(x))P(x, y)\, dx\, dy.$$

$E(h)$ is also called true error and always lies between 0 and 1. The classifier or predictor h is usually chosen from a function set $\mathcal{H}$. For example, $\mathcal{H}$ is determined by the architecture of a neural network, and the allocated h is determined by the network parameters. Learning means selecting an h from the set $\mathcal{H}$ so as to minimize $E(h)$. This is often done by Empirical Risk Minimization (ERM). Let $\mathcal{S} = \{(x_1, y_1), ..., (x_n, y_n)\}$ be a so-called training set of n input samples x together with their labels y which are drawn i.i.d. from $P(x, y)$. The empirical risk is defined as the average loss on this training set

$$E_{\mathcal{S}}(h) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, h(x_i))$$

and is also called training error. Learning via ERM chooses an $h \in \mathcal{H}$ that minimizes $E_{\mathcal{S}}(h)$. However, $E(h)$ and $E_{\mathcal{S}}(h)$ may deviate. This deviation is called the generalization gap.

If the generalization gap is small, a small training error provides a good solution with a small true error. Statistical learning theory based on VC-dimension [11] or Rademacher complexity [12] provides uniform bounds for the generalization gap. The generalization gap becomes small with increasing n for each $h \in \mathcal{H}$. The objective is to completely rule out "bad" solutions with large generalization gaps, at least with high probability. However, such bounds do not exist in highly over-parameterized regimes. There are always $h \in \mathcal{H}$ left with large generalization gaps. Nevertheless, in practice learning takes place and ERM provides good solutions that generalize well also in these over-parameterized regimes. There have

J. Martinetz is with the Machine Learning Group, Technical University Berlin, Berlin, Germany e-mail: j.martinetz@tu-berlin.de.

T. Martinetz is with the Institute of Neuro- and Bioinformatics, University of Lübeck, Lübeck, Germany e-mail: thomas.martinetz@uni-luebeck.de.

Manuscript received ... ; revised ... .

been theoretical and empirical attempts to understand this "mystery". It has been argued, that this might be due to implicit regularization of (stochastic) gradient descent [13], [14], [15], [16], [17], [18]. Further, a number of novel algorithmic-dependent uniform generalization bounds based on norm or compression measures have been supposed for explanation [19], [20], [21], [22], [23], [24], [25], [26], [27]. The Neural Tangent Kernel (NTK) with its linearization at initialization gives further insights for very wide neural networks [28], [29], [30], [31], [32], [33], [34]. Another approach applies the concepts of algorithmic stability [35], [36], [37], [38]. However, skepticism remains regarding whether these bounds can sufficiently explain the observed generalization in over-parameterized regimes. In [39], the authors conjecture that novel complexity measures might be necessary to achieve appropriate generalization bounds, presenting a comprehensive empirical study. Furthermore, [40] even argues that uniform convergence bounds might fundamentally not be the right approach to obtain non-vacuous bounds in over-parameterized scenarios.

As "bad" solutions cannot be ruled out in over-parameterized regimes, we conjecture that within the set of ERM solutions, the proportion of "bad" classifiers nevertheless remains small. This hypotheses offers a potential alternative explanation for the enigma of good generalization in over-parameterized regimes and corresponds to and supports the empirical studies in [41]. If the fraction of "bad" classifiers is small among the global minima with zero training error, the ERM algorithm is more likely to converge to a favorable solution rather than an unfavorable one. This perspective introduces a novel angle, challenging the conventional wisdom in machine learning that typically assumes the prevalence of "bad" solutions in highly over-parameterized regimes, and that it needs appropriate regularization to converge to good solutions. In this paper, we explore our conjecture, that in the over-parameterized regime bad solutions become rare in the solution set, and study this perspective mathematically in Section III and experimentally in Section IV.

II. Preliminaries, Notations and Definitions

We assume $\mathcal{H} = \mathcal{H}_{\mathcal{W}}$ to be parameterized on a compact set $\mathcal{W} \subseteq \mathbb{R}^N$ with non-zero Lebesgue measure, and each $w \in \mathcal{W}$ determines a classifier function $h_w \in \mathcal{H}_{\mathcal{W}}$. If a neural network is used as classifier, then w are the weights of the neural network. For each w we obtain a true error $E(h_w) = E(w)$. We assume $E(w)$ to be integrable on $\mathcal{W}$ and $\mathcal{H}_{\mathcal{W}}$ to have a large but finite VC-dimension. $E_{min} = \min_{w\in\mathcal{W}} E(w)$ denotes the minimum true error of the classifier function set $\mathcal{H}_{\mathcal{W}}$ on the given classification problem. We work with the following subsets of $\mathcal{W}$ with $\varepsilon \geq 0$:

$$\begin{aligned} \mathcal{W}_\varepsilon &: \text{ set of all } w \in \mathcal{W} \text{ with } E(w) \geq E_{min} + \varepsilon \\ \mathcal{W}(\mathcal{S}) &: \text{ set of all } w \in \mathcal{W} \text{ with } E_{\mathcal{S}}(w) = 0 \\ \mathcal{W}_\varepsilon(\mathcal{S}) &: \text{ set of all } w \in \mathcal{W}_\varepsilon \text{ with } E_{\mathcal{S}}(w) = 0 \end{aligned}$$

$\mathcal{W}(\mathcal{S})$ is the set of training error zero solutions for a given training set $\mathcal{S}$. In the over-parameterized regime where the training set size $n = |\mathcal{S}|$ is small compared to the complexity of the classifier, we assume $\mathcal{W}(\mathcal{S})$ to be of non-zero measure for almost all $\mathcal{S}$ with $|\mathcal{S}| = n$. Only "almost all", since there might be special cases of $\mathcal{S}$ which do not allow a training error zero solution, however, we assume them to be of measure zero, i.e., to have probability zero to occur in the random sampling process. This assumption could be taken as a definition for over-parameterization. Hence, the ERM algorithm ends up in one of these global minima with training error zero[1]. $\mathcal{W}_\varepsilon(\mathcal{S})$ is what we call the set of "bad" global minima with true errors larger than $E_{min} + \varepsilon$. We have $\mathcal{W}_\varepsilon(\mathcal{S}) \subseteq \mathcal{W}(\mathcal{S}) \subseteq \mathcal{W}$. Further, we use the notations $\Omega = |\mathcal{W}|$, $\Omega_\varepsilon = |\mathcal{W}_\varepsilon|$, $\omega(\mathcal{S}) = |\mathcal{W}(\mathcal{S})|$ and $\omega_\varepsilon(\mathcal{S}) = |\mathcal{W}_\varepsilon(\mathcal{S})|$ for the size (volume) of these parameter sets. In the following the fractions

$$\begin{aligned} g_\varepsilon = \frac{\Omega - \Omega_\varepsilon}{\Omega} &: \text{ fraction of "good" classifiers in } \mathcal{W} \\ \phi_\varepsilon(\mathcal{S}) = \frac{\omega_\varepsilon(\mathcal{S})}{\omega(\mathcal{S})} &: \text{ fraction of "bad" classifiers in } \mathcal{W}(\mathcal{S}) \end{aligned}$$

are important. g_ε as the fraction of "good" classifiers within $\mathcal{W}$ is a measure for the "appropriateness" of $\mathcal{H}$ for the given classification problem. It is a measure for the bias of the classifier function set towards the given classification problem. $\phi_\varepsilon(\mathcal{S})$ is the main property we are looking at. If $\phi_\varepsilon(\mathcal{S})$ is small, the ERM algorithm should more likely end up in a good solution. We show that under certain conditions $\phi_\varepsilon(\mathcal{S})$ is very likely to become small with increasing n and relate it to g_ε.

A. Density of Classifiers (DOC)

For quantifying the parameter volume of good and bad classifiers we introduce the "density of classifiers" $D(E)$ at true error E defined such that

$$\Omega_\varepsilon = \int_{E_{min}+\varepsilon}^{1} D(E)\, dE \quad \text{for each} \quad 0 \leq \varepsilon \leq 1 - E_{min}. \tag{1}$$

$D(E)dE$ is the volume of parameters $w \in \mathcal{W}$ with $E \leq E(w) \leq E + dE$. For $0 \leq E < E_{min}$ we set $D(E) = 0$.

In Fig. 1 we illustrate the density of classifiers (DOC) for two scenarios. On the left (A) we see the DOC of a neural network from our experiment in Section 4.1. For one million network weights $w \in \mathcal{W}$ uniformly chosen from $\mathcal{W}$ their corresponding true error $E(w)$ is determined (details in Section 4.1). The DOC in A shows the distribution of the true errors. The part left of the red line illustrates $g_\varepsilon = \int_0^{E_{min}+\varepsilon} D(E)\, dE/\Omega$, the fraction of good classifiers with a true error smaller than $E_{min}+\varepsilon$. On the right (B) we show the DOC in case of a binary classification problem with random class labels (each class with equal probability). For each w, the true error is $E(w) = 0.5$ and, hence, the DOC is a peak at $E = 0.5$.

III. Results

For $\mathcal{S}$ randomly drawn from $P(x, y)^n$, the parameter set size $\omega_\varepsilon(\mathcal{S})$ of "bad" global minima is a random variable with $0 \leq \omega_\varepsilon(\mathcal{S}) \leq \Omega$. The first and main Theorem, for which the proof is given in the Appendix, looks at its mean value:

[1]Convergence to global minima of ERM algorithms such as gradient descent or its variants is a topic on its own rights. However, in highly over-parameterized settings convergence to zero training error is usually not a problem.

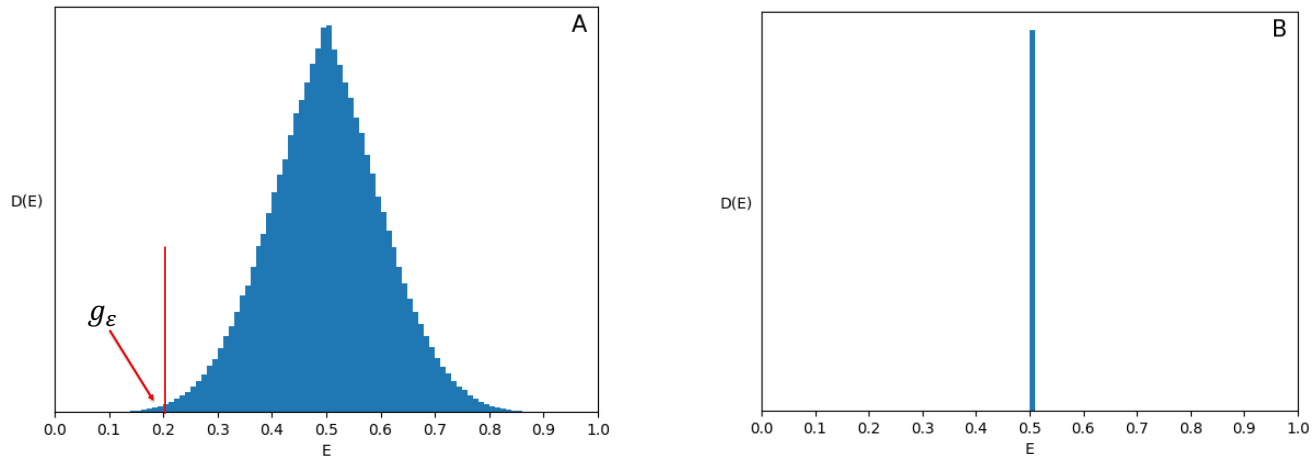

Fig. 1: The density of dlassifiers (DOC) for a neural network from our experiments (A) and a binary classification problem with random labels (B). In (A) the DOC is shown as a histogram of the true error for one million randomly chosen network weights. g_ε gives the fraction of weights with true errors lower than 0.2. In a binary classification problem with random labels (B), for each classifier w the true error $E(w)$ is always 0.5 and, hence, the DOC is a peak at $E = 0.5$. Note, that any values on the y-axis depend on the normalization of $D(E)$ and, therefore, are omitted. We will see that only the shape of $D(E)$ is important.

Theorem 1. *For each* $0 \leq \varepsilon \leq 1 - E_{min}$

$$\begin{aligned} \langle \omega_\varepsilon(\mathcal{S}) \rangle_\mathcal{S} &= \int_{\mathcal{W}_\varepsilon} (1 - E(w))^n \, dw \qquad (2) \\ &= \int_{E_{min}+\varepsilon}^{1} (1-E)^n D(E) \, dE \\ &\leq \Omega_\varepsilon (1 - (E_{min} + \varepsilon))^n. \end{aligned}$$

Theorem 1 tells us, that the mean parameter set size (parameter volume) of "bad" global minima decreases to zero exponentially fast with increasing training set size n. However, this might not suffice for $\phi_\varepsilon(\mathcal{S})$ to become small, since the mean parameter set size (volume) of ALL global minima, $\langle \omega(\mathcal{S}) \rangle_\mathcal{S}$, decreases exponentially fast as well. $\langle \omega(\mathcal{S}) \rangle_\mathcal{S}$ is given by (2) for $\varepsilon = 0$. Either it converges to zero, e.g. if $E_{min} > 0$, or to the mean set size of solutions with true error zero in case $E_{min} = 0$. If the classification problem is separable, this set size can be of non-zero measure.

However, the mean parameter set size of ALL global minima converges much slower than the mean set size of "bad" global minima and, hence, the mean set size of "bad" global minima will become very small relative to the mean set size of all global minima. The following corollary quantifies it (proof in the Appendix):

Corollary 1. *For each* $0 \leq \varepsilon < 1 - E_{min}$

$$\begin{aligned} \frac{\langle \omega_\varepsilon(\mathcal{S}) \rangle_\mathcal{S}}{\langle \omega(\mathcal{S}) \rangle_\mathcal{S}} &= \frac{\int_{E_{min}+\varepsilon}^{1} (1-E)^n D(E) \, dE}{\int_0^1 (1-E)^n D(E) \, dE} \qquad (3) \\ &\leq \frac{\Omega_\varepsilon}{\Omega} \frac{1}{(1 - g_{\varepsilon/2}) + g_{\varepsilon/2} \, e^{\frac{\varepsilon}{2} n}} \qquad (4) \\ &\leq \frac{1}{g_{\varepsilon/2}} \, e^{-\frac{\varepsilon}{2} n} \qquad \text{if} \quad g_{\varepsilon/2} > 0. \qquad (5) \end{aligned}$$

The mean set size of "bad" global minima relative to mean set size of all global minima diminishes at least exponentially fast with n. Note, that the prefactor in (5) is determined solely by the inverse of $g_{\varepsilon/2}$, i.e., by the fraction of "good" classifiers within $\mathcal{W}$ (better than $\varepsilon/2$) and, hence, does not depend on n. The larger the fraction of good classifiers to begin with, i.e., the better the bias of the classifier set, the smaller the prefactor. This fraction does not necessarily have to increase with Ω, e.g., with the number of parameters. We will demonstrate this later in our experiments. The exponent is linear in ε.

Corollary 1 tells us, that the mean set size of "bad" global minima is small relative to the mean set size of all global minima. In a concrete setting with a given training set $\mathcal{S}$, $\phi_\varepsilon(\mathcal{S}) = \omega_\varepsilon(\mathcal{S})/\omega(\mathcal{S})$, i.e., the given fraction of "bad" global minima should be small relative to the given fraction of all global minima. In general

$$\langle \phi_\varepsilon(\mathcal{S}) \rangle_\mathcal{S} = \left\langle \frac{\omega_\varepsilon(\mathcal{S})}{\omega(\mathcal{S})} \right\rangle_\mathcal{S} \neq \frac{\langle \omega_\varepsilon(\mathcal{S}) \rangle_\mathcal{S}}{\langle \omega(\mathcal{S}) \rangle_\mathcal{S}},$$

but with the following corollary we get a condition for the l.h.s. to be smaller than the r.h.s., which would be sufficient:

Corollary 2. *If and only if* $\phi_\varepsilon(\mathcal{S})$ *and* $\omega(\mathcal{S})$ *do not correlate negatively, then*

$$\begin{aligned} \langle \phi_\varepsilon(\mathcal{S}) \rangle_\mathcal{S} &\leq \frac{\langle \omega_\varepsilon(\mathcal{S}) \rangle_\mathcal{S}}{\langle \omega(\mathcal{S}) \rangle_\mathcal{S}}. \\ &\leq \frac{\int_{E_{min}+\varepsilon}^{1} (1-E)^n D(E) \, dE}{\int_0^1 (1-E)^n D(E) \, dE}. \qquad (6) \end{aligned}$$

Equality is given if and only if $\phi_\varepsilon(\mathcal{S})$ *and* $\omega(\mathcal{S})$ *do not correlate.*

Corollary 2 directly follows from its condition, since then $\langle \phi_\varepsilon(\mathcal{S}) \omega(\mathcal{S}) \rangle_\mathcal{S} - \langle \phi_\varepsilon(\mathcal{S}) \rangle_\mathcal{S} \langle \omega(\mathcal{S}) \rangle_\mathcal{S} \geq 0$, and from the definition of $\phi_\varepsilon(\mathcal{S})$ we have $\phi_\varepsilon(\mathcal{S}) \omega(\mathcal{S}) = \omega_\varepsilon(\mathcal{S})$.

With the Markov-inequality the probability, that $\phi_\varepsilon(\mathcal{S})$ is larger than a $\gamma > 0$, is given by

$$\begin{aligned} Prob\{\phi_\varepsilon(\mathcal{S}) \geq \gamma\} &\leq \frac{1}{\gamma} \frac{\int_{E_{min}+\varepsilon}^1 (1-E)^n D(E)\, dE}{\int_0^1 (1-E)^n D(E)\, dE} \\ &\leq \frac{1}{\gamma g_{\varepsilon/2}}\, e^{-\frac{\varepsilon}{2}n}. \end{aligned}$$

The probability, that the fraction $\phi_\varepsilon(\mathcal{S})$ of "bad" ERM solutions within the set of all ERM solutions is not small (smaller than γ) for a given $\mathcal{S}$, decreases exponentially fast to zero with the number of training data n. "Bad" solutions become rare with high probability. It is not the absolute number (density) of classifiers, which determines the r.h.s, but only the shape of $D(E)$, i.e., the normalized $D(E)$. The shape of $D(E)$ does not necessarily depend on the number of parameters of our classifier functions.

A. Mean true error

In the following we derive an expression for E_n, the mean true error over all global minima $\mathcal{W}_\mathcal{S}$ and all $\mathcal{S}$. For a given $\mathcal{S}$, we define $Q_\mathcal{S}(E)$ such that

$$\phi_\varepsilon(\mathcal{S}) = \int_{E_{min}+\varepsilon}^1 Q_\mathcal{S}(E)\, dE \quad \text{for each} \quad 0 \leq \varepsilon \leq 1 - E_{min}. \tag{7}$$

$Q_\mathcal{S}(E)$ is the normalized density of solutions with true error E within the set of training error zero solutions $\mathcal{W}_\mathcal{S}$. In concrete terms, for a given $\mathcal{S}$, within the set of training error zero solutions the fraction of solutions with true errors between E and $E + dE$ is given by $Q_\mathcal{S}(E)\, dE$. This fraction is higher for large $Q_\mathcal{S}$, indicating a higher likelihood of encountering solutions with true errors where the density $Q_\mathcal{S}(E)$ is high.

The mean true error $E_\mathcal{S}$ over all global minima of a given $\mathcal{S}$ is then given by

$$E_\mathcal{S} = \int_0^1 E\, Q_\mathcal{S}(E)\, dE.$$

With $Q_n(E) = \langle Q_\mathcal{S}(E) \rangle_\mathcal{S}$ we obtain from (7)

$$\langle \phi_\varepsilon(\mathcal{S}) \rangle_\mathcal{S} = \int_{E_{min}+\varepsilon}^1 Q_n(E)\, dE \;\text{ for each }\; 0 \leq \varepsilon \leq 1 - E_{min}, \tag{8}$$

and the mean true error E_n over all global minima $\mathcal{W}_\mathcal{S}$ and all $\mathcal{S}$ is accordingly given by

$$E_n = \int_0^1 E\, Q_n(E)\, dE. \tag{9}$$

If (and only if) Corollary 2 with equality in (6) is valid for each $0 \leq \varepsilon \leq 1 - E_{min}$, then comparing (6) and (8) gives

$$Q_n(E) = \frac{(1-E)^n D(E)\, dE}{\int_0^1 (1-E)^n D(E)\, dE},$$

and with (9) we obtain for the mean true error

$$E_n = \int_0^1 \frac{E(1-E)^n D(E)}{\int_0^1 (1-E)^n D(E)\, dE}\, dE. \tag{10}$$

In this case the mean true error over all ERM solutions is exactly determined by the density of classifiers $D(E)$. Again, it is not the absolute number (density) of classifiers, which determines E_n, but only the shape of $D(E)$.

In case that equality in (6) is not valid, we have to extend Corollary 2. For each $\varepsilon, \varepsilon'$ with $0 \leq \varepsilon < \varepsilon' \leq 1 - E_{min}$ we denote by $\mathcal{W}_{\varepsilon<\varepsilon'}(\mathcal{S})$ the set of all $w \in \mathcal{W}(\mathcal{S})$ with $E_{min} + \varepsilon \leq E(w) \leq E_{min} + \varepsilon'$ and with $\omega_{\varepsilon<\varepsilon'}(\mathcal{S})$ its size. Since $\omega_{\varepsilon<\varepsilon'}(\mathcal{S}) = \omega_\varepsilon(\mathcal{S}) - \omega_{\varepsilon'}(\mathcal{S})$ and with (3) we obtain

$$\frac{\langle \omega_{\varepsilon<\varepsilon'}(\mathcal{S}) \rangle_\mathcal{S}}{\langle \omega(\mathcal{S}) \rangle_\mathcal{S}} = \frac{\int_{E_{min}+\varepsilon}^{E_{min}+\varepsilon'} (1-E)^n D(E)\, dE}{\int_0^1 (1-E)^n D(E)\, dE}.$$

With $\phi_{\varepsilon<\varepsilon'}(\mathcal{S}) = \omega_{\varepsilon<\varepsilon'}(\mathcal{S})/\omega(\mathcal{S})$ and analog to Corollary 2, if and only if $\phi_{\varepsilon<\varepsilon'}(\mathcal{S})$ and $\omega(\mathcal{S})$ do not correlate negatively for each $0 \leq \varepsilon < \varepsilon' \leq 1 - E_{min}$, then

$$\begin{aligned} \langle \phi_{\varepsilon<\varepsilon'}(\mathcal{S}) \rangle_\mathcal{S} &\leq \frac{\int_{E_{min}+\varepsilon}^{E_{min}+\varepsilon'} (1-E)^n D(E)\, dE}{\int_0^1 (1-E)^n D(E)\, dE} \\ &\quad \text{for each} \qquad 0 \leq \varepsilon < \varepsilon' \leq 1 - E_{min}. \end{aligned}$$

Since $\phi_{\varepsilon<\varepsilon'}(\mathcal{S}) = \phi_\varepsilon(\mathcal{S}) - \phi_{\varepsilon'}(\mathcal{S})$ and with (8)

$$\begin{aligned} \langle \phi_{\varepsilon<\varepsilon'}(\mathcal{S}) \rangle_\mathcal{S} &= \int_{E_{min}+\varepsilon}^{E_{min}+\varepsilon'} Q_n(E)\, dE \\ &\quad \text{for each} \qquad 0 \leq \varepsilon < \varepsilon' \leq 1 - E_{min}. \end{aligned}$$

But then

$$Q_n(E) \leq \frac{(1-E)^n D(E)\, dE}{\int_0^1 (1-E)^n D(E)\, dE}$$

and with (9)

$$E_n \leq \int_0^1 \frac{E(1-E)^n D(E)}{\int_0^1 (1-E)^n D(E)\, dE}\, dE. \tag{11}$$

Interestingly, in our experiments in the next section E_n is not only bounded by (11) but in most cases exactly determined by (10).

B. Discussion of results

The central question is: Under what conditions does Corollary 2 hold, meaning when do $\phi_\varepsilon(\mathcal{S})$ and $\omega(\mathcal{S})$ not correlate negatively? The proportion of "bad" global minima, $\phi_\varepsilon(\mathcal{S})$, should not have a tendency of becoming smaller for larger global minima set sizes $\omega(\mathcal{S})$.

We take a look at the random variables $0 < \omega(\mathcal{S}) \leq \Omega$ and $0 \leq \omega_\varepsilon(\mathcal{S}) \leq \Omega_\varepsilon$ generated by random drawings of $\mathcal{S}$ and their joint probability distribution $P(\omega, \omega_\varepsilon)$. With $P(\omega, \omega_\varepsilon) = P(\omega_\varepsilon|\omega)P(\omega)$ we can define $\bar{\phi}_\varepsilon(\omega)$ and $\bar{\omega}_\varepsilon(\omega)$ according to

$$\begin{aligned} \bar{\phi}_\varepsilon(\omega) &= \int_0^{\Omega_\varepsilon} \frac{\omega_\varepsilon}{\omega}\, P(\omega_\varepsilon|\omega)\, d\omega_\varepsilon \\ &= \frac{1}{\omega} \int_0^{\Omega_\varepsilon} \omega_\varepsilon\, P(\omega_\varepsilon|\omega)\, d\omega_\varepsilon \\ &= \frac{\bar{\omega}_\varepsilon(\omega)}{\omega}. \end{aligned}$$

If $\bar{\phi}_\varepsilon(\omega)$ is monotonically non-decreasing with ω, then

$$\begin{aligned} \langle \phi_\varepsilon(\mathcal{S})\omega(\mathcal{S}) \rangle_\mathcal{S} - \langle \phi_\varepsilon(\mathcal{S}) \rangle_\mathcal{S} \langle \omega(\mathcal{S}) \rangle_\mathcal{S} \\ = \langle \phi_\varepsilon \omega \rangle_{\omega_\varepsilon,\omega} - \langle \phi_\varepsilon \rangle_{\omega_\varepsilon,\omega} \langle \omega \rangle_{\omega_\varepsilon,\omega} \\ = \left\langle \bar{\phi}_\varepsilon(\omega)\omega \right\rangle_\omega - \left\langle \bar{\phi}_\varepsilon(\omega) \right\rangle_\omega \langle \omega \rangle_\omega \\ \geq 0, \end{aligned}$$

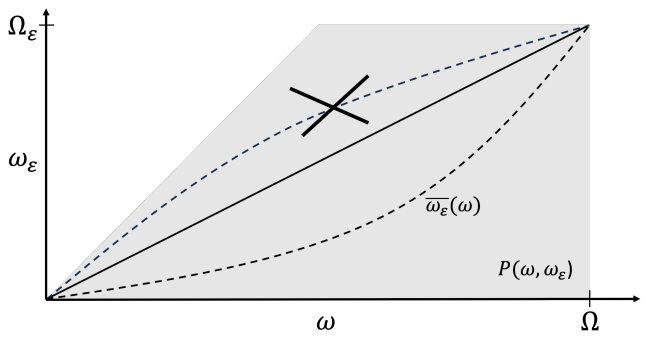

Fig. 2: Illustration of the structure of the distribution $P(\omega, \omega_\varepsilon)$ of the two random variables ω and ω_ε. For each $\mathcal{S}$ the corresponding pair $(\omega, \omega_\varepsilon)$ always lies in the grey shaded area. $\bar{\omega}_\varepsilon(\omega)$ goes to zero for $\omega \to 0$ and to Ω_ε for $\omega \to \Omega$. $\bar{\omega}_\varepsilon(\omega)$ never lies strictly above the straight line from the origin to $(\Omega, \Omega_\varepsilon)$, hence, it is never concave. If it is convex, then Corollary 2 is valid.

which is sufficient for Corollary 2 to be valid.

When is this the case, that $\bar{\phi}_\varepsilon(\omega)$ is monotonically non-decreasing with ω? In Fig. 2 we illustrate some scenarios. With a random $\mathcal{S}$, the corresponding pair $(\omega, \omega_\varepsilon)$ always falls into the grey area, since always $\omega_\varepsilon \leq \Omega_\varepsilon$ and $\omega_\varepsilon \leq \omega$. For $\omega \to 0$, $\bar{\omega}_\varepsilon(\omega) \to 0$, and for $\omega \to \Omega$, $\bar{\omega}_\varepsilon(\omega) \to \Omega_\varepsilon$. A simple course of $\bar{\omega}_\varepsilon(\omega)$ between these two points would be convex or concave, in the sense that the first derivative is monotone. However, the case that $\omega_\varepsilon(\omega)$ is strictly above the line given by $\omega_\varepsilon = (\Omega_\varepsilon/\Omega)\,\omega$, which includes being strictly concave, can be ruled out, since otherwise

$$\begin{aligned}
\frac{\langle \omega_\varepsilon(\mathcal{S})\rangle_\mathcal{S}}{\langle \omega(\mathcal{S})\rangle_\mathcal{S}} &= \frac{\langle \omega_\varepsilon\rangle_{\omega_\varepsilon,\omega}}{\langle \omega\rangle_{\omega_\varepsilon,\omega}} \\
&= \frac{1}{\langle \omega\rangle_\omega}\int_0^\Omega \bar{\omega}_\varepsilon(\omega)\,P(\omega)d\omega \\
&> \frac{1}{\langle \omega\rangle_\omega}\int_0^\Omega \frac{\Omega_\varepsilon}{\Omega}\,\omega\,P(\omega)d\omega \\
&> \frac{\Omega_\varepsilon}{\Omega},
\end{aligned}$$

which contradicts (4) of Corollary 1. But if $\bar{\omega}_\varepsilon(\omega)$ is convex, then per definition of convexity

$$\frac{\bar{\omega}_\varepsilon(\omega)}{\omega} = \bar{\phi}_\varepsilon(\omega)$$

is monotonically non-decreasing in ω and, hence, Corollary 2 is valid.

Thus, the trajectory of $\bar{\omega}_\varepsilon(\omega)$ must at least exhibit a complexity beyond that of a simple convex shape for Corollary 2 to be rendered invalid. While a higher complexity cannot be ruled out, it would not yet be sufficient and raise intriguing research questions. Our experiments in the subsequent section indeed consistently affirm the validity of Corollary 2.

IV. Experiments

To experimentally verify our theoretical results, we study experimental setups in which we can determine $D(E)$, $\omega(\mathcal{S})$ and $\phi_\varepsilon(\mathcal{S})$ by random sampling. We take multi-layer-perceptrons with leaky ReLUs (10% leakiness) and no bias. The output of the classifier is then given by

$$\mathbf{y} = ReLU(W_K ReLU(W_{K-1}(...ReLU(W_1\mathbf{x}))))$$

with W_k, $k = 1, ..., K$ as the weight matrix of layer k, $\mathbf{x} \in \mathbb{R}^D$ as the input vector of dimension D and $\mathbf{y} \in \mathbb{R}^2$ as the output vector, in our case for binary classification. The ReLU of the output layer with the largest output determines the predicted class membership for the given input $\mathbf{x}$. In this setting, it is easy to see and well known that the class membership is invariant to the lengths of the weight vectors due to the positive homogeneity of ReLUs as well as leaky ReLUs (for $s \geq 0$, $ReLU(sx) = sReLU(x)$). Thus, each classifier function is determined by a weight vector (comprising all weights of the network) of unit length. $\mathcal{W}$ is given by the unit sphere in dimension $\mathbb{R}^N$, with N as the number of weights of the network. In our experiments, we will sample uniformly from this unit sphere. With the leakiness of the ReLUs, we avoid too many trivial solutions with many "dead" ReLUs.

A. Synthetic data set

For the first experiment we take a synthetic data set of two isotropic Gaussians in 10 dimensions. The Gaussian of class A is centered at $(+1, 0, ..., 0)$ and of class B at $(-1, 0, ..., 0)$. Both Gaussians have a variance of 0.5 and, hence, overlap with 2.28%. This is the minimal possible classification error E_{min}. We apply three different networks. The smallest has one hidden layer with ten hidden units, which leads to 120 weights. The next has again one hidden layer, but 100 hidden units, hence, with $1,200$ ten times as many weights. The third network has ten hidden layers with ten units each, hence, $1,020$ weights.

The first row of Fig. 3 shows the results with the smallest network. On the left we see its density of classifiers $D(E)$ as a histogram of 100 bins (A1). We determined $D(E)$ by randomly sampling weight vectors from a uniform probability distribution on the 120-dimensional unit hypersphere. For each weight vector the true error of the corresponding classifier is measured with a balanced test set consisting of $10,000$ randomly chosen test data from the two classes. The result is put into the corresponding bin. This was repeated $1,000,000$ times. As expected, $D(E)$ is symmetric and has its peak at $E = 0.5$, and good classifiers with a small true error are rare within $\mathcal{W}$, i.e., on the hypersphere.

In the middle (B1) we see the distribution of the test (true) errors at randomly found global minima of randomly chosen training sets. For each n, the distribution of the test error along the y-axis corresponds to the mean normalized density of global minima $Q_n(E)$. We obtained this by taking a randomly chosen training data set, and then randomly sampled weight vectors from the unit hypersphere until we found one with training error zero on this training set. Then we measured the corresponding true error (red dot) with the test data set and then took the next randomly chosen training data set. For each n this was repeated $1,000$ times ($1,000$ red dots for each n). The box-plots show, that already for $n = 30$ more than 75% of the training error zero solutions provide test (true) errors of

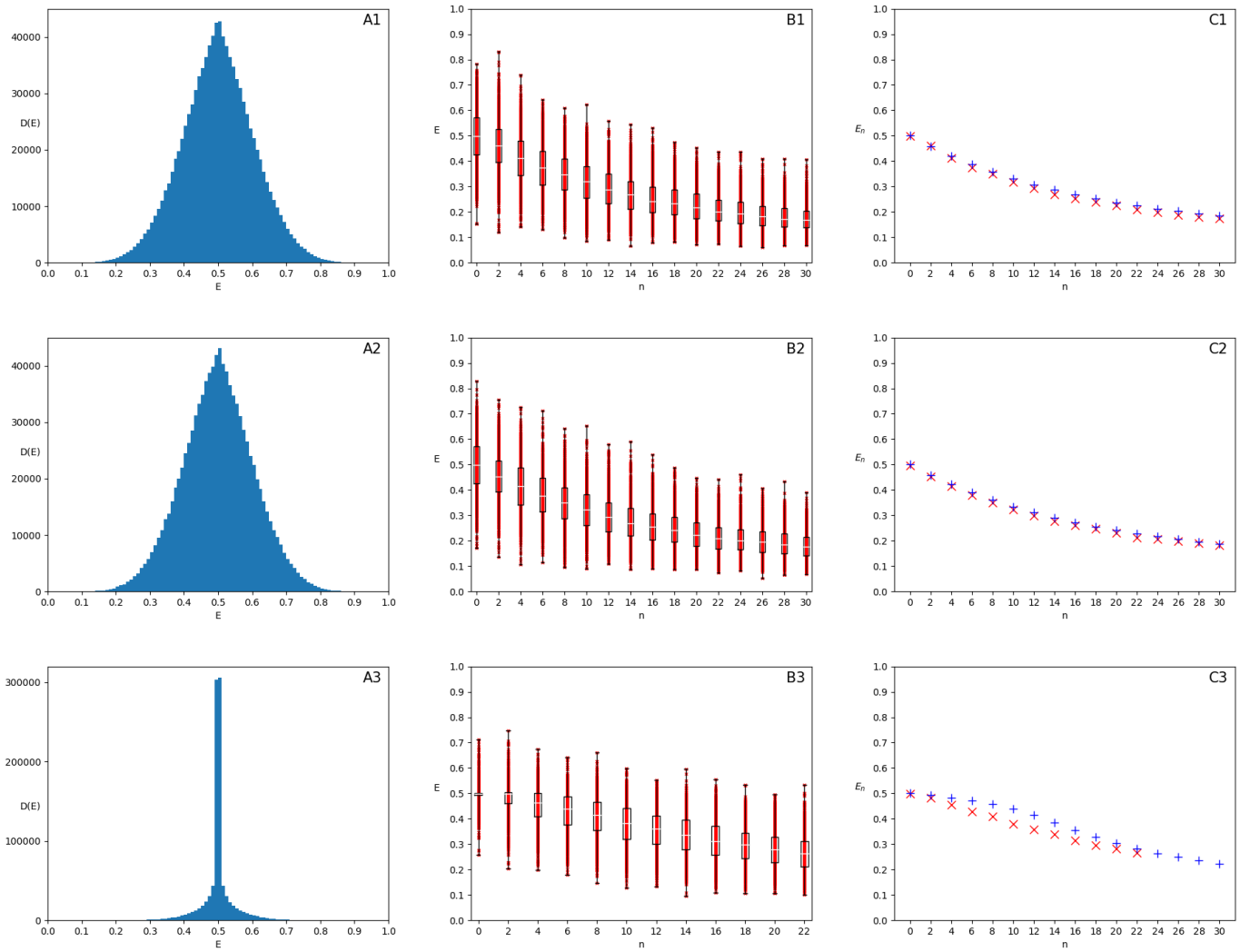

Fig. 3: Experiments with two Gaussians in 10 dimensions, classified by three different networks from top to bottom. The shape of the density of classifiers $D(E)$ (left column) of the first two networks hardly deviate, in spite of 10 times more weights in A2 than in A1. The column in the middle shows the distribution of test errors of training error zero solutions for increasing n. The average fraction of "bad" solutions converges to zero in these highly over-parameterized scenarios. In the right column the average of the test errors from the middle column (red x) are compared with the values from our bound determined by the $D(E)$s in the left column (blue crosses). In all three cases, bound (11) is valid, and in C1 and C2 it is even tight.

less than 0.2, i.e., for $n = 30$ we have $\langle \phi_{\varepsilon=0.2}(\mathcal{S}) \rangle = 0.25$. In only one of the $1,000$ trials there is a test error slightly above 0.4. "Bad" global minima are quickly becoming rare on the unit hypersphere.

On the right (C1) we show the average test error (red x) for each n, the average over the $1,000$ red dots from (B1). This corresponds to E_n of the l.h.s. of (10). The r.h.s. we can determine with the measured $D(E)$ from the left column, with the integral as sums over the bins. The results are shown by the blue crosses. Red and blue crosses match surprisingly well. This shows that bound (11) and, hence, also bound (6) is valid, indicating that there is no negative correlation. Even more, rather equality is given, which means there is no correlation at all.

The second row of Fig. 3 shows the results of the network with one hidden layer and 100 hidden units. This network has ten times as many weights as the previous network with 10 hidden units. Nevertheless, the shape of the density of classifiers $D(E)$ (A2) hardly deviates from the $D(E)$ of the much smaller network (A1). Also the distribution of the test (true) errors of training errors zero solutions (B1 and B2) are more or less identical. As for the ten times smaller network, already for $n = 30$ in about 75% of the cases one ends up with a test error of less than 0.2, and in none of the $1,000$ trials there is a test error above 0.4. Accordingly, also Figs. C1 and C2 look identical. Again, the red and blue crosses match in C2 and, hence, bound (6) is tight and there is no correlation.

In the third row of Fig. 3 we see the density of classifiers $D(E)$ for the network with ten hidden layers with ten neurons each and $1,020$ weights (A3). The large peak at $E = 0.5$ stems

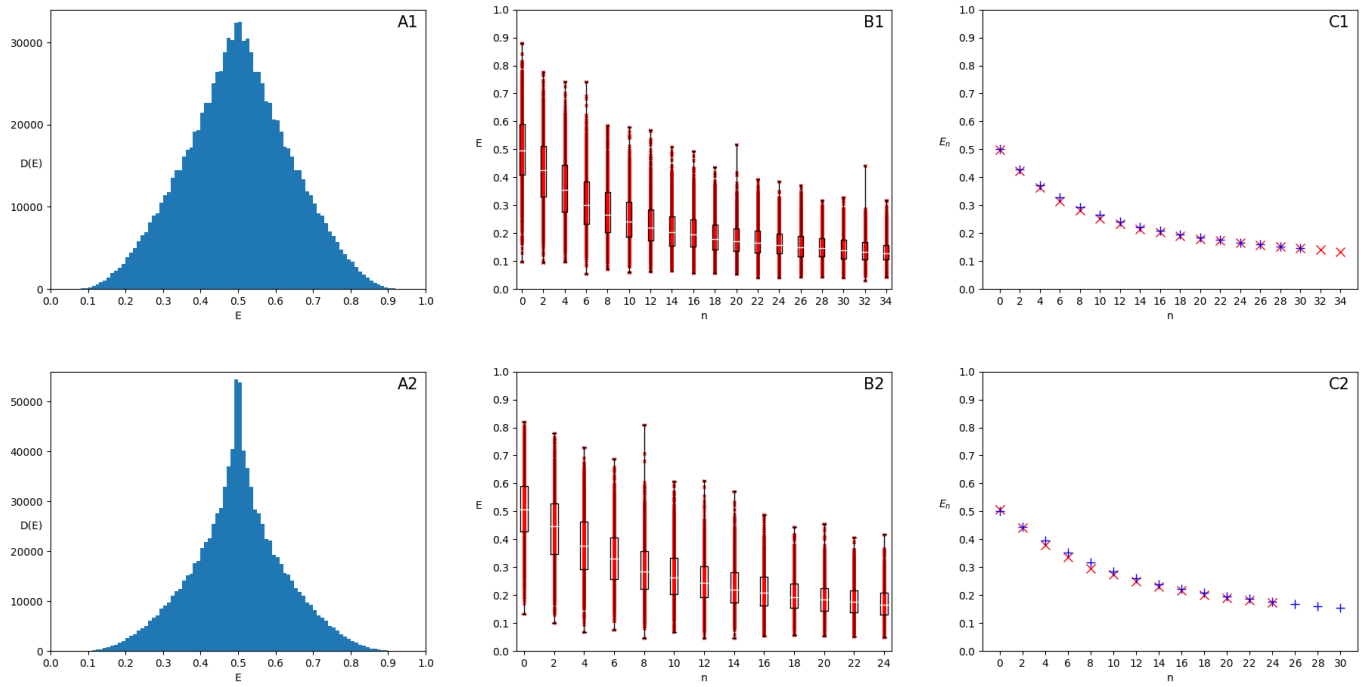

Fig. 4: Classification of the two digits 1 and 2 of the MNIST data set with two output ReLUs (first row) and with one hidden layer with 10 hidden ReLUs and two output ReLUs (second row). Left (A1, A2) the density of classifiers $D(E)$ and in the middle (B1, B2) the distribution of the test errors (red dots) of training error zero solutions for increasing number of training data n. On the right (C1, C2) the average of the test errors (red x) compared with values of the bound (blue crosses) derived from the respective $D(E)$. Bound (11) holds and again is even tight in both cases.

from weight configurations where the classifier output indicates always the same class, independent from the input. The classifiers h which correspond to these weight configurations are ruled out immediately as soon as training data from both classes occur, since then these h do not provide zero training error and do not belong to the global minimum. With $D(E)$ having a different shape as in the two cases before, also Fig. B3 looks differently now. It converges more slowly, and the peak of $D(E)$ resolves with increasing n, as can be seen from the increasing box. For example, for $n = 2$ in half of the training data instances the training data are from the same class and there still are solutions from the peak at $E = 0.5$. We stopped with $n = 22$, since the time to find zero training error solutions by random sampling started to take hours on a 13th Gen Intel(R) Core(TM) i7-13700K, GPU: NVIDIA® GeForce RTX™ 4090. It is important to note that the primary runtime bottleneck is the generation of random weight vectors, a task handled by the CPU. Since no gradient-based training of weights is involved, fast GPUs do not provide any significant performance benefit in our case. In Fig. C3 the red and blue crosses do not match for every n, but the blue crosses always lie above the red ones. Bound (6) is valid, but now for some n there is a positive correlation between $\phi_\varepsilon(\mathcal{S})$ and $\omega(\mathcal{S})$.

B. Data set from MNIST

In Fig. 4 we show the results of the same experiments with real data. We took the digits 1 and 2 of the MNIST data set as the two classes to be classified. Each digit image has a size of 28×28 pixels and thus the input dimension is 784. We used the same network structure as before, in one case with no hidden layer, i.e., only two leaky output ReLUs, one for each class, and in the other case a one-hidden layer MLP with 10 hidden leaky ReLUs and two leaky output ReLUs. In the first case, the classifier has $1,568$ weights, and in the second case $7,860$ weights. The training data sets were randomly drawn from a set of $6,000$ images of each digit, respectively. These images were taken from the MNIST training set. For testing, 900 images of each digit were taken from the MNIST test set.

The results look very much like in Fig. 3 for the synthetic data. In the first row of Fig. 4, the results without hidden layer and $1,568$ weights and in the second row the results with ten hidden units and $7,860$ weights are shown. From left to right we see the density of classifiers $D(E)$, the distribution of test errors at training error zero for different n, and the average test errors compared to values of the bound (6). Again, the bound holds and even equality is given, i.e., there is no correlation between the fraction of "bad" global minima $\phi_\varepsilon(\mathcal{S})$ and the size of the set of global minima $\omega(\mathcal{S})$.

In both scenarios, the classifiers are highly over-parameterized for these small training data sets, but nevertheless good solutions quickly dominate. For both networks, already for $n = 24$ about 75% of all test errors are below 0.2. Instead of a fifty-fifty chance of incorrect predictions, 75% of the solutions with zero training error make fewer than one mistake in five cases.

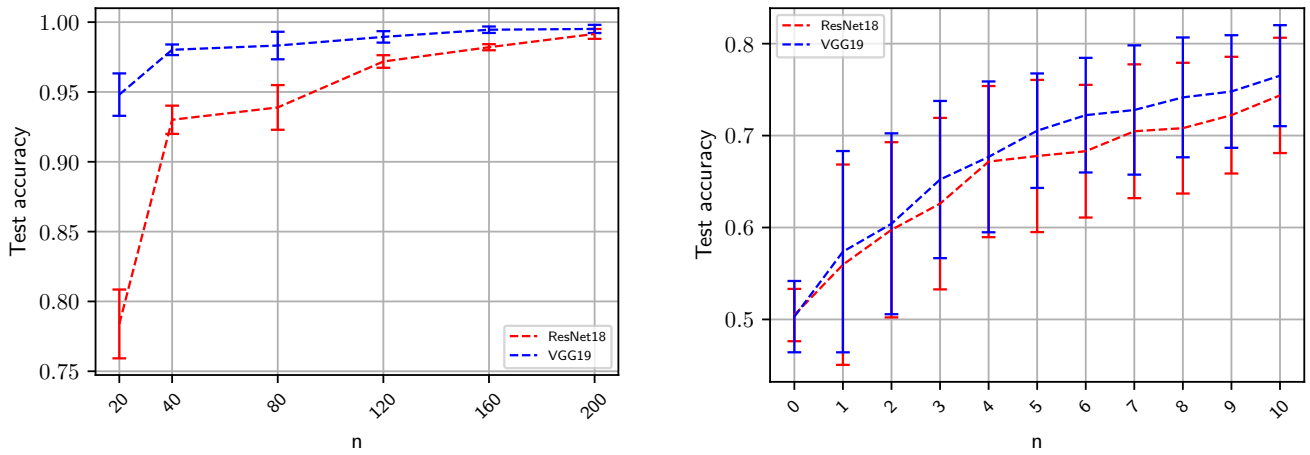

Fig. 5: The left plots are adapted from [10] and show the mean test accuracies (and variances) in classifying motorcycles and airplanes on images from the Caltech101 dataset with VGG19 and ResNet18, respectively. Both networks are trained with stochastic gradient descent up to training error zero. To the right we see the mean test accuracies and their variances of solutions with training error zero obtained by random sampling. Good solutions quickly dominate within the set of solutions with training error zero, and the more than ten times larger VGG19 converges faster than ResNet18, which corresponds to the results obtained with gradient descent (left).

C. VGG and ResNet and Caltech101

Finally, we take a look at very large Deep Networks. In [10] was shown that a VGG19 net [42] with 140 million weights is able to learn to discriminate airplanes and motorcycles on images from the Caltech101 dataset [43] up to 95% accuracy trained with only 20 examples from each class. This extremely over-parameterized network was initialized randomly and then trained by stochastic gradient descent up to training error zero without any regularization. 798 motorbike and 798 aircraft images were used with different training and test splits. Fig. 5A shows these results adapted from [10]. With 200 training data the test accuracy is almost 100% despite complete over-fitting and memorizing the training data.

We hypothesize that the above results can be explained by "bad" solutions being rare in the set of training error zero solutions. Therefore, an ERM algorithm like stochastic gradient descent should very likely end up in a good solution also without regularization. We test our hypotheses like in the previous sections. Again, due to the positive homogeneity of the VGG19 net each classifier function realized by VGG19 corresponds to a point on the unit hypersphere in the weight space. We uniformly draw random weights from this sphere, and if the corresponding classifier function has zero training error we determine its test accuracy. The training examples are drawn from 798 motorbike and aircraft images from Caltech101, always the same number for each class, analog to the experiments in Fig. 5A. The rest of the images are used for determining the test accuracy. Like in Fig. 5A we perform the same experiment also for ResNet18 [4], which has $11,2$ million parameters - more than ten times less than VGG19.

Fig. 5B shows the results for up to 10 samples per class (for larger n it became too computationally expensive). For each n we randomly drew random training sets and weights until we reached 100 zero training error solutions. Fig. 5B shows the mean of the corresponding test accuracies and their variances. To find zero training error solutions required several million trials for larger n. Interestingly, on average it required much more trials for ResNet18 than for VGG19. Obviously, the fraction of training error zero solutions within the weight space $\mathcal{W}$ is larger on average for VGG than for ResNet18. Already for $n = 10$ the training error zero solutions of VGG19 have a mean test accuracy of 75%. A rough extrapolation to $n = 20$ indeed leads to the range of VGG19 in Fig. 5A. Also ResNet18 being worse than VGG19 in spite of being ten times smaller corresponds to Fig. 5A. Both is in line with our conjecture.

V. Discussion

Experiments consistently demonstrate that highly over-parameterized classifiers can learn effectively even without explicit regularization and despite "memorizing" the training data up to zero training error. This contradicts traditional machine learning wisdom, rooted in learning theory based on uniform convergence, which typically requires (with high probability) a uniformly small generalization gap. This implies the complete absence of "bad" classifiers within the set of possible ERM solutions with zero training error. This stringent requirement necessitates substantial training data. However, it might be reasonable to allow for a small fraction of "bad" classifiers, rather than demanding zero, within all ERM solutions. As long as the proportion of "bad" classifiers is small, there remains a high probability for the ERM algorithm to discover a good solution, potentially achieved with significantly less training data.

We proved mathematically for classification tasks, that within the over-parameterized regime, the mean set size or parameter volume of "bad" ERM solutions diminishes at a much faster rate with the increase in the number of training samples compared to the mean set size of all possible ERM solutions. If the

fraction of "bad" classifiers within the set of ERM solutions does not exhibit a negative correlation with the volume of ERM solutions, the fraction of "bad" classifiers will be small with high probability.

We introduced the density of classifiers (DOC) at a true error E, which depends on the set of classifier functions and the classification problem. The DOC determines the bound for the fraction of "bad" classifiers and serves as the primary characteristic of a given classifier/problem case. Since its shape (i.e., the normalized DOC) is important, it does not necessarily depend on the classifier's capacity or its number of parameters. Increasing the absolute density values by expanding the capacity of the classifier function set may keep the normalized DOC invariant, or may even improve it, as observed in numerous experiments.

We experimentally validated our theoretical results using a synthetic dataset and a subset of MNIST with three and two different network architectures and sizes, respectively. Through random sampling, we determined the density of classifiers (DOC) and the distribution of the true error on the set of ERM solutions. Indeed, the theoretical bounds complied with the experiments, and in most cases the theoretical values even matched precisely with the measured mean errors from the experiments. The experiments have shown that the DOC is not necessarily dependent on the complexity and size of the classifier. Networks with ten times as many parameters had similar DOCs, leading to a comparable decay in the fraction of "bad" minima, consistent with our theoretical findings. Finally, we applied our framework to VGG19 and ResNet18, two Convolutional Neural Networks with millions of parameters. Again, the fraction of good solutions increases immediately already with a handful of training data, in line with the unexpected good generalization of these networks observed in highly over-parameterized settings and without explicit regularization.

Of course, (stochastic) gradient descent, being the standard Empirical Risk Minimization (ERM) algorithm for learning, does not necessarily converge to each of the global minima with equal probability. It is influenced by various factors such as initialization, error landscape structure, learning rates, and more. Consequently, unlike the random sampling in our experiments, the probability of landing in a "bad" solution is not precisely determined by the fraction of "bad" solutions. Nonetheless, for (stochastic) gradient descent to consistently converge to the tiny fraction of "bad" solutions, it would have to exhibit an "exponential" bias in favor of such solutions. Traditional machine learning wisdom previously assumed the opposite—that good solutions are rare and that substantial guidance, such as explicit or implicit regularization, is essential to locate them and avoid "bad" solutions. In this context, we present a novel perspective that may contribute to understanding the unexpected behavior of highly over-parameterized networks.

In our theoretical framework, we required the absence of a negative correlation between the fraction of "bad" classifiers within the set of global minima and the size of the set of global minima. We outlined several considerations concerning the joint distribution of these two quantities and their conditional means, suggesting that the absence of negative correlation appears to be more plausible than the reverse. In our experiments, we indeed observed no negative correlation, and even no correlation at all in most of the evaluated settings. However, since random sampling for finding a training error zero solution becomes very time-consuming with increasing training set size, in our experiments we were restricted to relatively small settings.

For a comprehensive theoretical understanding, it remains an open question under what conditions, concerning the mathematical structure of the classifier and the classification problem, there exists no negative or even no correlation. Such a correlation is a macroscopic structural characteristic of the classifier/problem setting, which might be influenced by high over-parameterization or specific aspects of the structure of neural networks. This intriguing question remains open for exploration in future research, offering an avenue for deeper insights into the behavior of highly over-parameterized classifiers.

APPENDIX

Proofs

Theorem 1:

Proof. For a given $w \in \mathcal{W}$, the loss $L(y, h_w(x))$ is a binomial random variable assuming the values 0 or 1 for training data drawn i.i.d. from $P(x, y)$. The probability that $L(y, h_w(x)) = 0$ for a (x, y) drawn from $P(x, y)$ is $1 - E(w)$. Hence, for a $\mathcal{S}$ randomly drawn from $P(x, y)^n$, the probability for h_w to be a consistent classifier with $L(y_i, h_w(x_i)) = 0$ for each $(x_i, y_i) \in \mathcal{S}$ is $(1 - E(w))^n$.

We define the indicator function $\mathbf{1}_{\mathcal{S}}(h_w)$, which is one for $w \in \mathcal{W}(\mathcal{S})$ and otherwise zero. For a fixed w and $\mathcal{S}$ randomly drawn from $P(x, y)^n$, the probability for $\mathbf{1}_{\mathcal{S}}(h_w)$ assuming the value one is $(1 - E(w))^n$. With this we obtain

$$\begin{aligned} \langle \omega_\varepsilon(\mathcal{S}) \rangle_{\mathcal{S}} &= \left\langle \int_{\mathcal{W}_\varepsilon} \mathbf{1}_{\mathcal{S}}(h_w)\, dw \right\rangle_{\mathcal{S}} \\ &= \int_{\mathcal{W}_\varepsilon} \langle \mathbf{1}_{\mathcal{S}}(h_w) \rangle_{\mathcal{S}}\, dw \\ &= \int_{\mathcal{W}_\varepsilon} (1 - E(w))^n\, dw. \end{aligned} \tag{12}$$

In the second step we interchanged integrals. This can be done, if for a random sequence $\mathcal{S}_1, ..., \mathcal{S}_m$ of training sets

$$\frac{1}{m} \sum_{i=1}^{m} \mathbf{1}_{\mathcal{S}_i}(h_w) \;\to\; \langle \mathbf{1}_{\mathcal{S}}(h_w) \rangle_{\mathcal{S}} \quad \text{uniformly in probability}$$

on $\mathcal{W}_\varepsilon$ for $m \to \infty$. But this is the case due to the finite VC-dimension of our classifier set $\mathcal{H}_{\mathcal{W}}$. $\mathcal{H}_{\mathcal{W}}$ loses the capacity to shatter $\mathcal{S}_1, \ldots, \mathcal{S}_m$ with m becoming sufficiently large and, hence, the Uniform Convergence Theorem [44] applies.

From the definition of $D(E)$ in (1), we have

$$\Omega_\varepsilon = \int_{\mathcal{W}_\varepsilon} dw = \int_{E_{min}+\varepsilon}^{1} D(E) dE,$$

which allows to change variables in (12) accordingly to obtain

$$\begin{aligned}\langle\omega_\varepsilon(\mathcal{S})\rangle_{\mathcal{S}} &= \int_{E_{min}+\varepsilon}^{1}(1-E)^n\,D(E)dE \\ &\leq (1-(E_{min}+\varepsilon))^n\int_{E_{min}+\varepsilon}^{1}D(E)dE \\ &= \Omega_\varepsilon(1-(E_{min}+\varepsilon))^n,\end{aligned}$$

which concludes the proof. □

Corollary 1:

Proof. We show that for any $a>1$ and $0\leq\varepsilon<1-E_{min}$

$$\begin{aligned}\frac{\langle\omega_\varepsilon(\mathcal{S})\rangle_{\mathcal{S}}}{\langle\omega(\mathcal{S})\rangle_{\mathcal{S}}} &\leq \frac{\Omega_\varepsilon}{\Omega}\frac{1}{(1-g_{\varepsilon/a})+g_{\varepsilon/a}\;e^{(1-1/a)\varepsilon n}} \\ &\leq \frac{1}{g_{\varepsilon/a}}\;e^{-(1-1/a)\varepsilon n}.\end{aligned}$$

Corollary 1 is the special case for $a=2$. The r.h.s. of course requires $g_{\varepsilon/a}>0$.

With Theorem 1 and dividing the integral in the denominator into two parts we have

$$\begin{aligned}\frac{\langle\omega_\varepsilon(\mathcal{S})\rangle_{\mathcal{S}}}{\langle\omega(\mathcal{S})\rangle_{\mathcal{S}}} &= \frac{\int_{E_{min}+\varepsilon}^{1}(1-E)^nD(E)\,dE}{\int_0^1(1-E)^nD(E)\,dE} \\ &= \frac{\int_{E_{min}+\varepsilon}^{1}(1-E)^nD(E)\,dE}{\int_0^{E_{min}+\varepsilon}(1-E)^nD(E)dE+\int_{E_{min}+\varepsilon}^{1}(1-E)^nD(E)dE} \\ &= \frac{1}{1+\frac{\int_0^{E_{min}+\varepsilon}(1-E)^nD(E)\,dE}{\int_{E_{min}+\varepsilon}^{1}(1-E)^nD(E)\,dE}} \\ &\leq \frac{1}{1+\frac{\int_0^{E_{min}+\varepsilon}(1-E)^nD(E)\,dE}{\Omega_\varepsilon(1-(E_{min}+\varepsilon))^n}}\end{aligned}$$

For any $a>1$ and dividing the integral into two parts, we obtain for the nominator

$$\begin{aligned}&\int_0^{E_{min}+\varepsilon}(1-E)^nD(E)\,dE \\ &= \int_0^{E_{min}+\varepsilon/a}(1-E)^nD(E)\,dE \\ &\quad+\int_{E_{min}+\varepsilon/a}^{E_{min}+\varepsilon}(1-E)^nD(E)\,dE \\ &\geq (\Omega-\Omega_{\varepsilon/a})(1-(E_{min}+\varepsilon/a))^n \\ &\quad+(\Omega_{\varepsilon/a}-\Omega_\varepsilon)(1-(E_{min}+\varepsilon))^n.\end{aligned}$$

But then with some basic transformations

$$\begin{aligned}\frac{\langle\omega_\varepsilon(\mathcal{S})\rangle}{\langle\omega(\mathcal{S})\rangle} &\leq \frac{1}{1+\frac{\Omega_{\varepsilon/a}-\Omega_\varepsilon}{\Omega_\varepsilon}+\frac{(\Omega-\Omega_{\varepsilon/a})(1-(E_{min}+\varepsilon/a))^n}{\Omega_\varepsilon(1-(E_{min}+\varepsilon))^n}} \\ &\leq \frac{\Omega_\varepsilon}{\Omega}\frac{1}{(1-g_{\varepsilon/a})+g_{\varepsilon/a}\;\frac{(1-(E_{min}+\varepsilon/a))^n}{(1-(E_{min}+\varepsilon))^n}}.\end{aligned}$$

For the factor in the denominator with n in the exponent we can write

$$\frac{(1-(E_{min}+\varepsilon/a))^n}{(1-(E_{min}+\varepsilon))^n}=e^{n(\ln(1-(E_{min}+\varepsilon/a))-\ln(1-(E_{min}+\varepsilon)))}.$$

With the Taylor-expansion $\ln(1-x)=-x-x^2/2-x^3/3-...$, it is easy to see that in the exponent

$$\ln(1-(E_{min}+\varepsilon/a))-\ln(1-(E_{min}+\varepsilon))\geq(1-1/a)\varepsilon,$$

which concludes the proof. □

Acknowledgement

We thank Christoph Linse for discussions and providing the graphs in Figure 5.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Neural Information Processing Systems*, vol. 25, 01 2012.
[2] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018.
[3] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, Feb. 2015.
[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
[5] P. Nakkiran, G. Kaplun, Y. Bansal, T. Yang, B. Barak, and I. Sutskever, "Deep double descent: Where bigger models and more data hurt," in *ICLR 2020*, 2020. Also appeared in Journal of Statistical Mechanics, 2021.
[6] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.
[7] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," 2017.
[8] M. Belkin, D. Hsu, S. Ma, and S. Mandal, "Reconciling modern machine-learning practice and the classical bias–variance trade-off," *Proceedings of the National Academy of Sciences*, vol. 116, p. 201903070, 07 2019.
[9] M. Loog, T. Viering, A. Mey, J. H. Krijthe, and D. M. J. Tax, "A brief prehistory of double descent," *Proceedings of the National Academy of Sciences*, vol. 117, no. 20, pp. 10625–10626, 2020.
[10] C. Linse and T. Martinetz, "Large neural networks learning from scratch with very few data and without explicit regularization," *15th International Conference on Machine Learning and Computing (ICMLC)*, 2023.
[11] V. N. Vapnik, *Statistical Learning Theory*. Wiley-Interscience, 1998.
[12] P. L. Bartlett and S. Mendelson, "Rademacher and gaussian complexities: Risk bounds and structural results.," *J. Mach. Learn. Res.*, vol. 3, pp. 463–482, 2002.
[13] B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro, "Exploring generalization in deep learning," in *NIPS*, 2017.
[14] A. Brutzkus, A. Globerson, E. Malach, and S. Shalev-Shwartz, "SGD learns over-parameterized networks that provably generalize on linearly separable data," in *International Conference on Learning Representations*, 2018.
[15] D. Soudry, E. Hoffer, M. S. Nacson, S. Gunasekar, and N. Srebro, "The implicit bias of gradient descent on separable data," *Journal of Machine Learning Research*, vol. 19, no. 70, pp. 1–57, 2018.
[16] K. Lyu and J. Li, "Gradient descent maximizes the margin of homogeneous neural networks," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia*, 2020.
[17] M. S. Advani, A. M. Saxe, and H. Sompolinsky, "High-dimensional dynamics of generalization error in neural networks," *Neural Networks*, vol. 132, pp. 428–446, 2020.
[18] G. Vardi, "On the implicit bias in deep-learning algorithms," *Commun. ACM*, vol. 66, p. 86–93, may 2023.
[19] B. Neyshabur, R. Tomioka, and N. Srebro, "Norm-based capacity control in neural networks," in *Proceedings of the 28th Conference on Learning Theory (COLT)*, vol. PMLR 40, pp. 1376–1401, 2015.
[20] P. Bartlett, D. Foster, and M. Telgarsky, "Spectrally-normalized margin bounds for neural networks," *Advances in Neural Information Processing Systems*, vol. 30, pp. 6241–6250, 2017.
[21] N. Golowich, A. Rakhlin, and O. Shamir, "Size-independent sample complexity of neural networks," in *COLT*, 2018.

[22] S. Arora, R. Ge, B. Neyshabur, and Y. Zhang, "Stronger generalization bounds for deep nets via a compression approach," in *Proceedings of the 35th International Conference on Machine Learning*, pp. 254–263, 2018.
[23] Y. Li and Y. Liang, "Learning overparameterized neural networks via stochastic gradient descent on structured data," in *Advances in Neural Information Processing Systems* (S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds.), vol. 31, Curran Associates, Inc., 2018.
[24] V. Nagarajan and Z. Kolter, "Deterministic PAC-bayesian generalization bounds for deep networks via generalizing noise-resilience," in *International Conference on Learning Representations*, 2019.
[25] B. Neyshabur, Z. Li, S. Bhojanapalli, Y. LeCun, and N. Srebro, "The role of over-parametrization in generalization of neural networks," in *International Conference on Learning Representations*, 2019.
[26] C. Wei and T. Ma, "Data-dependent sample complexity of deep neural networks via lipschitz augmentation," in *NeurIPS*, 2019.
[27] T. Liang, T. Poggio, A. Rakhlin, and J. Stokes, "Fisher-rao metric, geometry, and complexity of neural networks," in *Proceedings of Machine Learning Research*, pp. 888–896, 2019.
[28] A. Jacot, F. Gabriel, and C. Hongler, "Neural tangent kernel: Convergence and generalization in neural networks," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, p. 8580–8589, 2018.
[29] L. Chizat, E. Oyallon, and F. R. Bach, "On lazy training in differentiable programming," in *Neural Information Processing Systems*, 2018.
[30] Z. Allen-Zhu, Y. Li, and Y. Liang, "Learning and generalization in overparameterized neural networks, going beyond two layers," in *Neural Information Processing Systems*, 2018.
[31] S. Arora, S. S. Du, W. Hu, Z. Li, R. Salakhutdinov, and R. Wang, "On exact computation with an infinitely wide neural net," in *Neural Information Processing Systems*, 2019.
[32] S. Arora, S. S. Du, W. Hu, Z. Li, and R. Wang, "Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks," in *International Conference on Machine Learning*, 2019.
[33] J. Lee, L. Xiao, S. S. Schoenholz, Y. Bahri, R. Novak, J. Sohl-Dickstein, and J. Pennington, "Wide neural networks of any depth evolve as linear models under gradient descent*," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2020, p. 124002, dec 2020.
[34] H. Min, S. Tarmoun, R. Vidal, and E. Mallada, "On the explicit role of initialization on the convergence and implicit bias of overparametrized linear networks," *Proceedings of Machine Learning Research*, 2021.
[35] M. Hardt, B. Recht, and Y. Singer, "Train faster, generalize better: Stability of stochastic gradient descent," in *Proceedings of The 33rd International Conference on Machine Learning* (M. F. Balcan and K. Q. Weinberger, eds.), vol. 48 of *Proceedings of Machine Learning Research*, (New York, New York, USA), pp. 1225–1234, PMLR, 20–22 Jun 2016.
[36] W. Mou, L. Wang, X. Zhai, and K. Zheng, "Generalization bounds of sgld for non-convex learning: Two theoretical viewpoints," in *Proceedings of the 31st Conference On Learning Theory* (S. Bubeck, V. Perchet, and P. Rigollet, eds.), vol. 75 of *Proceedings of Machine Learning Research*, pp. 605–638, PMLR, 06–09 Jul 2018.
[37] Y. Lei, R. Jin, and Y. Ying, "Stability and generalization analysis of gradient methods for shallow neural networks," in *Advances in Neural Information Processing Systems* (A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, eds.), 2022.
[38] L. Oneto, S. Ridella, and D. Anguita, "Do we really need a new theory to understand over-parameterization?," *Neurocomput.*, vol. 543, jul 2023.
[39] Y. Jiang, B. Neyshabur, H. Mobahi, D. Krishnan, and S. Bengio, "Fantastic generalization measures and where to find them," in *International Conference on Learning Representations*, 2020.
[40] V. Nagarajan and J. Z. Kolter, "Uniform convergence may be unable to explain generalization in deep learning," in *Neural Information Processing Systems*, 2019.
[41] P. yeh Chiang, R. Ni, D. Y. Miller, A. Bansal, J. Geiping, M. Goldblum, and T. Goldstein, "Loss landscapes are all you need: Neural network generalization can be explained without the implicit bias of gradient descent," in *The Eleventh International Conference on Learning Representations*, 2023.
[42] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[43] F.-F. Li, M. Andreeto, M. Ranzato, and P. Perona, "Caltech 101," Apr 2022.
[44] V. N. Vapnik and A. Y. Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities," *Theory of Probability & Its Applications*, vol. 16, no. 2, pp. 264–280, 1971.

PLACE PHOTO HERE

Julius Martinetz Biography text here.

PLACE PHOTO HERE

Thomas Martinetz Biography text here.

UNIVERSITÄT ZU LÜBECK
INSTITUT FÜR
NEURO- UND BIOINFORMATIK

Universität zu Lübeck • Institut für Neuro- und Bioinformatik
Ratzeburger Allee 160 • 23562 Lübeck

Reviewer Comments, Author Responses and Manuscript Changes

Reviewer 1

Comments to the Author
The authors investigate the generalization capabilities of over-parameterized classifiers, where the number of parameters far exceeds the number of training examples. Typically, such conditions are expected to lead to overfitting; however, empirical observations suggest otherwise. The paper proposes that the rarity of "bad" solutions (solutions with zero training error but high true error) in the solution space under certain conditions might explain this phenomenon. This hypothesis is tested through mathematical formulations and experimental validation using synthetic data and subsets of well-known datasets like MNIST and Caltech101.

Strengths
Theoretical Novelty: The paper introduces fresh theoretical insights into the behavior of over-parameterized networks, challenging existing paradigms about generalization and overfitting.
Comprehensive Methodological Approach: The blend of theoretical derivations with empirical validations provides a robust argument for the proposed hypothesis.

Relevance and Implications: The findings have significant implications for the design and training of neural networks, particularly in how we understand and harness over-parameterization.

Thank you for your positive evaluation of our work. We very much appreciate that you consider it a valuable scientific contribution. In the following, we will address your helpful suggestions point-by-point.

Weaknesses and Suggestions for Improvement
Assumptions on the Uniformity of Solution Space: The theoretical model assumes a level of uniformity in the distribution of solutions that may not hold in practical scenarios with real-world data. Future work could focus on relaxing these assumptions or exploring non-uniform distributions of solutions.

Yes, this is an aspect which should be explored in future work. So far, we have been studying and measuring volumes within the weight space. For this purpose, it needs uniform sampling. However, as discussed in the "Discussion" section, in practice with a gradient based ERM algorithm there are many factors that result in non-uniform probabilities within the solution space. This is an important and intriguing aspect and we agree that this should be considered in the next phase of our research.

Scalability of Experimental Validation: The experimental validations, while insightful, are limited to relatively simple or controlled datasets. Testing the findings on larger, more complex datasets or in different real-world applications would provide deeper insights into the applicability of the theory.

We made first steps in this direction using subsets of MNIST and Caltech101. However, the primary challenge is the computational effort, especially with larger networks. As the amount of training data increases, finding training error zero solutions through random sampling becomes extremely time-consuming. The main bottleneck is the generation of random weight vectors, a task handled by the CPU. Since our experiments do not involve gradient-based training of weights, utilizing fast GPUs does not offer a significant advantage (we now mention this aspect in the manuscript). But we agree, it is

certainly an area for future work to further extend the experiments to larger and more complex data sets.

Detail on Negative Correlations: The paper discusses the absence of negative correlations between the fraction of bad classifiers and the volume of ERM solutions but does not thoroughly explore or validate this aspect empirically. More focused studies on this specific element could strengthen the theoretical framework.

We did not explicitly explore the absence of negative correlations experimentally, which is also not clear how to do it. Nevertheless, our experiments confirm the absence of negative correlations in Figs. 3 and 4 in column C, respectively. Our mathematical results indicate that the values of the blue and red crosses coincide, if and only if there is no correlation, and the values of the blue crosses are larger, if and only if there is a positive correlation. Since this is what we observe in our experiments, it empirically confirms the absence of negative correlations in the experimentally studied scenarios. We already addressed this important point in our manuscript, but now have revised the relevant paragraphs to clarify it further. But you are right, as we discussed in the last paragraph of the discussion, the absence of negative correlations is a cornerstone and should be an area of further theoretical and empirical studies.

Reviewer 2

Comments to the Author
The authors propose that over-parameterized neural networks generalize well because "bad" solutions are rare, amongst the set of global optima of DNN learning objectives. The authors do not explain "why" bad solutions are rare -- for example, is there something about ReLU activation functions, or particular DNN architectures, that makes such solutions rare. But they do support their hypothesis both analytically and experimentally. Before this paper is published, the authors should address the following concerns:

Thank you for your evaluation of our work. We greatly appreciate your comments and try to make the "why" clearer by going through your feedback point-by-point. Understanding how ReLUs or specific DNN architectures contribute to the rarity of bad solutions is certainly an important and interesting area for future research:

1) Page 1: "quantum leap in many real-world..." -- do the authors mean "quantum leap in performance, in many real-world..."

Thank you for this hint. You are right, this is not clearly expressed. We changed it accordingly.

2) "By enlarging... does not harm generalization." --> the authors should cite references to support this claim.

We added references.

3) "This is usually attributed to an appropriate regularization." -- cite a reference for this.

We added a reference.

4) "even after only a few handful of training samples" -- do you mean "even after training on only a handful of training samples?

You are right, this is not clearly expressed and we changed it accordingly.

5) "bad solutions... Interestingly, they hardly seem to occur in practice" -- cite a reference for this.

We added a reference.

6) "scepticism" --> "skepticism"

Thank you for pointing out this typo. We corrected it.

7) The authors should indicate whether their results can be extended beyond binary classification to both multi-class classification, as well as to regression (they only provide experimental results for binary classification...).

Yes, indeed, we should indicate that it applies to both binary as well as multi-class classification. We revised the manuscript accordingly to clarify it.

8) Is g_{epsilon/2} not a function of n? This should be clarified...

Thank you. We revised the text to make it clear.

9) "One will rather end up with a classifier at true errors where the density Q_S(E) is large." -- I did not understand this sentence -- I believe this description can be significantly improved...

Yes, you are right, this description needed improvement. Another reviewer also pointed out difficulties. We revised the paragraph and added further explanations to make the meaning of Q_S(E) clearer and more concrete.

10) "tendancy" --> "tendency"

Thank you for pointing out this typo. We corrected it.

11) "If phi bar epsilon(omega) is monotonically non-decreasing with omega, then..." -- but why should this be the case? The authors should justify this...

Thank you for this feedback. In fact, the following paragraph discusses the scenarios when this is the case and the assumption is justified. Obviously, this was not clear, so we revised the first sentence introducing the corresponding paragraph.

12) "If it is convex, than Corollary 2 is valid" --> "than" should be "then"

Thank you for pointing out this typo. We corrected it.

13) Is it just leaky ReLUs that satisfy positive homogeneity? Doesn't that also hold for standard ReLUs?

Yes, of course, positive homogeneity holds for both, standard and leaky ReLUs. We revised the corresponding sentence to clarify it.

14) 1.568 --> 1,568 and 7.8760 --> 7,860

Thank you, we corrected it.

15) Can the authors please try to explain why more trials are needed to find global minima for ResNet18 than for VGG19?

We added a sentence for explanation. It is an empirical observation but can be contextualized within the framework of our approach.

16) Figure 5 caption: "the more than ten times larger VGG19 converges faster than ResNet18, which corresponds to gradient descent" --> "corresponds to gradient descent": no idea why you are trying to say here -- this description should be improved.

Thank you for this feedback. The corresponding sentence was unclear and we revised it.

Reviewer 3

Comments to the Author
This is an interesting paper which demonstrates both mathematically and empirically that highly over-parameterised classifiers can obtain good generalisation, in contrast to conventional wisdom. The work is novel in that it presents a new theorem and two corollarys on neural network based classifiers. The work is significant in that it generates new insights for the neural network community. And both the mathematical analysis and empirical evaluation demonstrate an appropriate degree of rigour. The introduction is well-written and has a good narrative flow. The problem is described well.

Thank you very much for your positive assessment of our work and the constructive feedback. In the following we will go through your feedback point-by-point:

Not sure what is meant by this statement: “But still there is empirically based scepticism that these bounds already help [36] or that uniform convergence bounds are fundamentally the right approach [37].” Are you saying that given the empirical evidence we do not really need to have bounds for the generalisation gap?

Thank you for this feedback concerning this paragraph. We have revised it to make it clearer. What is meant is that so far the current bounds for the generalization gap in over-parameterized regimes are either vacuous or only valid in very special cases and still very loose. UNIFORM convergence bounds for the generalization gap - convergence of the generalization gap uniformly over the whole set of classifiers and being valid for any probability distribution - is probably not the right approach to obtain bounds which are non-vacuous for highly over-parameterized scenarios. We refer to the two cited papers which support this perspective.

Not sure what is meant: “we assume W(S) to be of non-zero measure for almost all S with |S| = n.”

Thank you for this important feedback. It is meant that we assume over-parameterization (relative to the training set size n) in the sense that for every training set S of size n drawn i.i.d. from the data distribution, there is always a set of weights with training error zero (“W(S) to be of non-zero measure”), except for eventually special cases of S but which have probability zero (“almost all S”). We revised the respective paragraph, since one of the preceding sentences probably made it unclear.

Would be good if the authors could state explicitly their contribution, particularly with respect to ref [7], at the end of the intro.

We added a sentence at the end of the introduction, what in principle we contribute with our paper, namely the conjecture and a mathematical and experimental exploration of this conjecture, that in the over-parameterized regime bad solutions become rare in the solution set. The details of our contribution are listed in the Abstract. The work of Ref [7] (now Ref [10]) was simply an empirical study on standard data sets with standard network architectures, that explicit regularization does not seem to be necessary to obtain good solutions with large networks and small training data sets. It should be clearer now.

In relation to Eqn. (6) not sure what it means to talk about true error for a given S? Once you have S then it becomes ERM?

Thank you for this valuable feedback. Another reviewer also pointed out difficulties around Eqn. (6) (now Eqn. (7)). We revised the paragraph and added further explanations to make the meaning of Eqn. (6) clearer and more concrete.

Figure 3. y-axis not labelled. Unusual way of labelling subfigures.

We labelled the y-Achses in all figures, not only Fig. 3.

Would be useful to understand what hardware platform was used for the experiments.

We used the following hardware platform: CPU: 13th Gen Intel(R) Core(TM) i7-13700K, GPU: NVIDIA® GeForce RTX™ 4090, RAM: 32 GB, SSD: 2TB (PCI). It is important to note that in our experiments, the primary runtime bottleneck is the generation of random weight vectors, a task handled by the CPU. Since our experiments do not involve gradient-based training of weights, fast GPUs do not provide any significant performance benefit. We address this aspect now in the manuscript.

For the experiments with VGG and ResNet how many test images were used?

798 motorbike and 798 aircraft images were used overall with different training and test splits. We added this information in the text.

The derivation of theorem 1 in the appendix should be more detailed. Would be good to get more insight into the 3rd step when changing the integration variable from over dw to dE. Use of a graphic would help illustrate. Similarly further detail should be given on the final step leading to the inequality.

Thank you for this hint. We have expanded the proof and clarified that the change of integration variables originates from the definition of D(E). To enhance the clarity of the final step, we have inserted an intermediate step in the derivation.