Good Classifiers are Abundant in the Interpolating Regime

Ryan Theisen
Dept. of Statistics
University of California, Berkeley
theisen@berkeley.edu

Jason M. Klusowski
Dept. of Operations Research
and Financial Engineering
Princeton University
jason.klusowski@princeton.edu

Michael W. Mahoney
ICSI and Dept. of Statistics
University of California, Berkeley
mmahoney@stat.berkeley.edu

Abstract

Within the machine learning community, the widely-used uniform convergence framework has been used to answer the question of how complex, over-parameterized models can generalize well to new data. This approach bounds the test error of the *worst-case* model one could have fit to the data, but it has fundamental limitations. Inspired by the statistical mechanics approach to learning, we formally define and develop a methodology to compute precisely the full distribution of test errors among interpolating classifiers from several model classes. We apply our method to compute this distribution for several real and synthetic datasets, with both linear and random feature classification models. We find that test errors tend to concentrate around a small *typical* value ε^*, which deviates substantially from the test error of the worst-case interpolating model on the same datasets, indicating that "bad" classifiers are extremely rare. We provide theoretical results in a simple setting in which we characterize the full asymptotic distribution of test errors, and we show that these indeed concentrate around a value ε^*, which we also identify exactly. We then formalize a more general conjecture supported by our empirical findings. Our results show that the usual style of analysis in statistical learning theory may not be fine-grained enough to capture the good generalization performance observed in practice, and that approaches based on the statistical mechanics of learning may offer a promising alternative.

Proceedings of the 24[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

1 INTRODUCTION

The phenomenon of good generalization in highly over-parameterized models, including neural networks, has largely eluded theoretical understanding. Recently, however, progress has been made towards understanding over-parameterization in several simpler settings. Important examples include the variety of results demonstrating "double descent" phenomena in linear regression (Belkin et al., 2019, Bartlett et al., 2020, Hastie et al., 2019, Dereziński et al., 2019) (and, in particular, how it is essentially a consequence of a transition between two different phases of learning (Liao et al., 2020)), nearest neighbors models (Xing et al., 2019), and binary classification (Chatterji and Long, 2020, Deng et al., 2020). These results are typically derived by defining a specific estimator (e.g., the least-norm estimator in linear regression), and carefully examining its test risk. This approach presents a challenge when extending these analyses to the setting of neural networks, where no such estimator can easily be defined. In these situations, almost all results rely, in one way or another, on the framework of *uniform convergence*; that is, results which bound a quantity of the form

$$\varepsilon_{\text{unif}} \coloneqq \sup_{f \in \mathcal{F}} \left| \widehat{\mathcal{E}}_n(f) - \mathcal{E}(f) \right|, \tag{1}$$

where $\mathcal{F}$ is a given function class, $\widehat{\mathcal{E}}_n$ is the training error on a dataset of n points, and $\mathcal{E}$ is the population error.

Recently, it has been drawn into question whether this approach is fine-grained enough to capture the good generalization properties observed in deep learning (Martin and Mahoney, 2017, Nagarajan and Kolter, 2019). One issue that arises when using the uniform convergence framework is that for any given training set $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, and a sufficiently complex function class $\mathcal{F}$, the worst-case estimator $f \in \mathcal{F}$ fitting the training data may indeed perform quite poorly—thus dooming

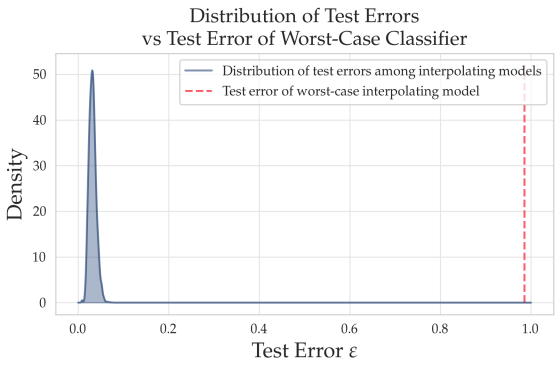

Figure 1: Test error distribution of MNIST 0 vs 1 interpolating classifiers, using $N = 1000$ random ReLU features, with $n = 500$ training samples, as well as test error of worst-case interpolating classifier. Here, for illustrative purposes, we plot the PDF (fit from a histogram using a kernel density estimate); in the remainder of the paper, we instead plot the CDFs, which can be more accurately estimated.

quantities like (1)—even if we are extremely unlikely to encounter such models in practice. One line of work has attempted to tackle this problem by studying the implicit biases of the algorithms used to train modern machine learning models (Gunasekar et al., 2018, Ma et al., 2020, Soudry et al., 2018) (by using what may be called implicit regularization in non-exact approximation algorithms (Mahoney, 2012)). Still, such results are mostly limited to simplified settings, and a comprehensive understanding of the relationship between optimization and generalization remains elusive.

In another line of work (Wu and Zhu, 2017, Choromanska et al., 2015), it has been observed that, at least in practice for deep networks, it is not particularly important which model we obtain at the end of training; most models tend to have roughly the same test error. Reconciling this phenomenon with the worst-case theory must then require one of a few things to be true: i) that most models have nearly worst-case test error; ii) that models with nearly worst-case error are very rare; or iii) that worst-case bounds are simply too loose to capture the actual worst-case error. In this paper, we investigate these possibilities rigorously in the setting of linear and random feature classification, and we find that worst-case models with very high test error do in fact exist, but that they are exceedingly rare.

Our approach builds conceptually on several old ideas originating out of the statistical physics literature. (Such a perspective, while less common in statistical learning theory today, has a long history (Martin and Mahoney, 2017, Seung et al., 1992, Watkin et al., 1993, Haussler et al., 1996, Engel and Van den Broeck, 2001).) Rather than studying the *worst-case* estimator $f \in \mathcal{F}$, the statistical mechanics approach seeks to understand the behavior of the *typical* function f. This typicality can be characterized in a number of ways. A natural measure, from the statistical physics perspective, would be the *entropy* (or log density of states), which captures the number of models at any given test error value. Analyses of learning problems have been conducted using the entropy method in a variety of simplified settings, including the case of finite $\mathcal{F}$ as well as linear classification under various simplifying assumptions on the data (Haussler et al., 1996, Opper and Haussler, 1991, Engel and Van den Broeck, 2001). Similar approaches have also been used to demonstrate the existence of phase transitions in learning behavior in logistic regression (Candes and Sur, 2018) and generalized linear models (Barbier et al., 2019). In the deep learning literature, (Choromanska et al., 2015) used the theory of spin glasses to argue that poor local minima on the training surface are rare. While insightful (and often technically impressive), many of these theoretical results rely on very specific assumptions on the data generating process, and hold only in the asymptotic regime.

In this paper, we study the behavior of test errors on real-world datasets used in practice, in a non-asymptotic regime, and without any assumptions on the data generating process. To do this, in Section 2, we formally define and develop a methodology to compute precisely the full distribution of test errors among interpolating classifiers from several model classes. In Sections 3 and 4, we then apply this methodology to compute these distributions for several real and synthetic datasets, and for both linear and random feature classification models, respectively. We furthermore develop a method to estimate the worst-case test errors of these classification models on the same datasets. Our investigation yields the following key insights:

1. Good classifiers are abundant: an overwhelming proportion of interpolating models have very small test error, relative to the worst-case error.
2. Test errors tend to concentrate: as the size of models grow, test errors concentrate sharply around a critical value ε^*.
3. There exist worst-case classifiers that are very poor: much worse than the typical classifier.

These findings are illustrated in Figure 1.

To understand these observations mathematically, in Section 5, we provide theoretical results in a simple setting in which we characterize the full (asymptotic) dis-

tribution of test errors, and we show that these indeed concentrate around a value ε^*, which we also identify exactly. We then formalize a more general conjecture, supported by our empirical findings, which we hope will motivate further research. Finally, in Section 6, we offer some concluding thoughts, and provide several promising directions for future work. Proofs and additional empirical results can be found in the technical report version of this paper (Theisen et al., 2020).

2 EFFICIENTLY COMPUTING THE DISTRIBUTION OF TEST ERRORS FOR INTERPOLATING CLASSIFIERS

2.1 Notation and Setup

We begin with some notation that will be used throughout the paper.

We will consider the setting of binary classification, and we denote a training dataset by $S_n = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$, with samples $\mathbf{x}_i \in \mathbb{R}^d$ and labels $y_i \in \{-1, 1\}$. We let $\mathcal{F}$ be a class of functions $f : \mathbb{R}^d \to \{-1, 1\}$, and we define the *version space* to be the following subset of $\mathcal{F}$:

$$\mathsf{VS}(S_n) = \{f \in \mathcal{F} : f(\mathbf{x}_1) = y_1, \dots, f(\mathbf{x}_n) = y_n\}. \quad (2)$$

That is, the version space is the set of "interpolating" functions, i.e., those which perfectly fit the dataset S_n. Note that if $\mathcal{F}$ is a linear family, then one element of the version space is the max-margin solution. We also use $\mathbb{P}$ to denote a probability measure defined over $\mathcal{F}$. We use $S_{\text{test}} = \{(\mathbf{x}_{n+1}, y_{n+1}), \dots, (\mathbf{x}_{n+m}, y_{n+m})\}$ to denote a set of m testing points, and $\Pr_{\mathbf{x},y}$ to denote a testing distribution over the data $(\mathbf{x}, y)$. Using these, we define the empirical and population testing errors:

$$\mathcal{E}_m(f) = \frac{1}{m}\sum_{h=1}^{m} \mathbb{1}(-y_{n+h} f(\mathbf{x}_{n+h}) > 0), \quad (3)$$

$$\mathcal{E}(f) = \Pr_{\mathbf{x},y}(-yf(\mathbf{x}) > 0). \quad (4)$$

With these definitions in place, we can now formally define the test error distribution of interpolating classifiers.

Definition 1. *Given a function class $\mathcal{F}$, a measure $\mathbb{P}$ over $\mathcal{F}$, and a training set S_n, let*

$$R_{n,m}(\varepsilon) := \frac{\mathbb{P}(\{\mathcal{E}_m(f) \le \varepsilon\} \cap \mathsf{VS}(S_n))}{\mathbb{P}(\mathsf{VS}(S_n))}, \quad (5)$$

and

$$R_n(\varepsilon) := \frac{\mathbb{P}(\{\mathcal{E}(f) \le \varepsilon\} \cap \mathsf{VS}(S_n))}{\mathbb{P}(\mathsf{VS}(S_n))}. \quad (6)$$

That is, the quantities $R_{n,m}(\varepsilon)$ and $R_n(\varepsilon)$ are the cumulative distribution functions (CDFs) of the errors $\mathcal{E}_m$ and $\mathcal{E}$, conditioned on perfectly fitting the training data. Intuitively, these quantities measure the fraction of interpolating classifiers $f \in \mathsf{VS}(S_n)$ that have test error at most ε.

2.2 Efficient Estimation of $R_{n,m}$

An advantage of our definition of $R_{n,m}(\varepsilon)$ is that it is defined only relative to fixed training and testing sets, S_n and S_{test}. This means that, at least in principle, $R_{n,m}(\varepsilon)$ can be computed exactly (without explicit knowledge of the training and testing distributions). To do this naïvely would require computing the ratio of two (in general very small) high-dimensional volumes, which would be costly and also lead to issues with numerical instability. Instead, a natural estimator for $R_{n,m}(\varepsilon)$ can be generated as follows: sample $\hat{f}_1, \dots, \hat{f}_M \sim \mathbb{P}(\cdot \mid \mathsf{VS}(S_n))$, and compute

$$\widehat{R}_{n,m}(\varepsilon) = \frac{1}{M}\sum_{j=1}^{M} \mathbb{1}(\mathcal{E}_m(\hat{f}_j) \le \varepsilon).$$

Standard Gilvenko-Cantelli-type results can be used to guarantee that $\sup_\varepsilon |R_{n,m}(\varepsilon) - \widehat{R}_{n,m}(\varepsilon)| = O(\frac{1}{\sqrt{M}})$. Hence, assuming we have the ability to sample from $\mathbb{P}(\cdot \mid \mathsf{VS}(S_n))$, the distribution $R_{n,m}(\varepsilon)$ can be estimated to arbitrary precision.

For the remainder of this section, we show how we can generate samples $\hat{f} \sim \mathbb{P}(\cdot \mid \mathsf{VS}(S_n))$ for any function class of the form $\mathcal{F}_\phi = \{f(\mathbf{x}) = \text{sign}(\mathbf{w}^\top \phi(\mathbf{x})) : \mathbf{w} \in \mathbb{R}^N\}$, where $\phi : \mathbb{R}^d \to \mathbb{R}^N$ is any mapping. In this paper, we will address the following important examples:

$$\phi(\mathbf{x}) = \mathbf{x}, \qquad \text{(linear classification)}$$
$$\phi(\mathbf{x}) = \sigma(\mathbf{U}\mathbf{x}). \qquad \text{(random features)}$$

Notice that for these classes of functions, a probability measure $\mathbb{P}$ over $\mathcal{F}$ is simply a distribution over $\mathbb{R}^N$. Throughout this paper, we will assume that $\mathbb{P}$ is the uniform distribution on the sphere $\mathbb{S}^{N-1} = \{\mathbf{w} \in \mathbb{R}^N : \|\mathbf{w}\| = 1\}$. This choice is made so as to obtain results that are agnostic to the choice of optimization algorithm: since any reasonable measure on the sphere will be absolutely continuous with respect to $\mathbb{P}$, we do not expect our main conclusions to be qualitatively changed by choosing a different base distribution. For the sake of computation, it will be convenient to make use of the equivalence (up to scaling) of the uniform distribution with the Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$, which is a consequence of the spherical symmetry of the Gaussian.

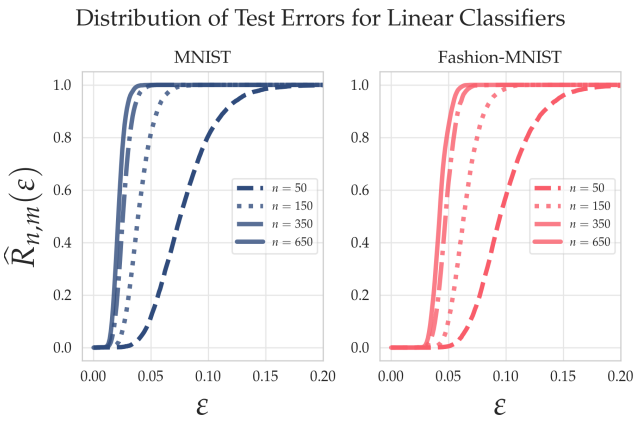

Figure 2: Estimated test error distribution $\widehat{R}_{n,m}(\varepsilon)$ for interpolating linear classifiers on the MNIST (0 vs 1) dataset (**blue**) and FASHION-MNIST (shirt vs pants) dataset (**red**).

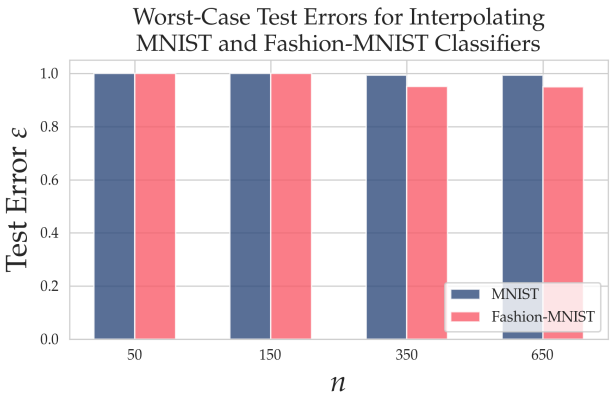

Figure 3: Test errors of interpolating classifiers with fit to n "good" training samples and $n_b = (d-1)-n$ "bad" training samples. The classifiers constructed here have extremely poor test set performance, in contrast to results shown in Figure 2.

Let us define the function

$$\mathcal{L}_n(\mathbf{w}) = \prod_{i=1}^{n} \mathbb{1}(y_i \mathbf{w}^\top \phi(\mathbf{x}_i) \geq 0), \tag{7}$$

and notice that $\mathbb{P}(\cdot \mid \mathsf{VS}(S_n)) = \mathbb{P}(\cdot \mid \mathcal{L}_n = 1)$. Therefore, we are interested in drawing samples from a linearly constrained Gaussian distribution. Fortunately, the recent work (Gessner et al., 2020) developed the LIN-ESS algorithm (an extension of Elliptical Slice Sampling (Murray et al., 2010)) specifically for this purpose. Using traditional Monte Carlo methods, this task would be computationally infeasible in high dimensions, since if we naïvely drew samples from $\mathbb{P}$ and rejected those not lying in the domain $\{\mathcal{L}_n(\mathbf{w}) = 1\}$, then drawing a reasonable number of samples could take an exponential amount of time. In contrast, LIN-ESS is able to exploit special properties of the linear constraints $y_i \mathbf{w}^\top \phi(\mathbf{x}_i) \geq 0$ to draw samples *without rejection*. In particular, in our setup, LIN-ESS can be used to generate samples $\widehat{\mathbf{w}}_1, \ldots, \widehat{\mathbf{w}}_M \sim \mathbb{P}(\cdot \mid \mathcal{L}_n = 1)$, which we can then use to compute the estimator $\widehat{R}_{n,m}(\varepsilon)$. As is the case with most MCMC algorithms, LIN-ESS is only guaranteed to produce independent samples from the posterior $\mathbb{P}(\cdot \mid \mathcal{L}_n = 1)$ asymptotically; we mitigate this issue in practice by using 1,000 warm-up samples, and keeping only every 10th sample thereafter.

3 LINEAR CLASSIFICATION

In this section, we compute the estimated test error distributions $\widehat{R}_{n,m}(\varepsilon)$ and $\widehat{R}_n(\varepsilon)$ on both real benchmark data as well as illustrative synthetic data, for the class $\mathcal{F}_{\text{LIN}} = \{f(\mathbf{x}) = \text{sign}(\mathbf{w}^\top \mathbf{x}) : \mathbf{w} \in \mathbb{R}^d\}$ of linear classifiers.

3.1 Evaluation on Image Datasets

For our first set of evaluations, we compute $\widehat{R}_{n,m}(\varepsilon)$ for high-dimensional image datasets used in modern machine learning. In particular, we focus on the MNIST and FASHION-MNIST datasets, which consist of images in $d = 784$ dimensional space. Thus, throughout this section, we only consider values of $n < 784$. Since we are specialized to the binary classification setting, we focus on the MNIST 0 vs 1 task, and on the shirt vs pants task for FASHION-MNIST. For both of these tasks, the data has been centered and scaled, so as to have mean 0 and variance 1.

In Figure 2, we plot the $\widehat{R}_{n,m}(\varepsilon)$ for various values of n. For each of the plots in this section, estimators $\widehat{R}_{n,m}(\varepsilon)$ are formed with $M = 10,000$ samples from $\mathbb{P}(\cdot \mid \mathcal{L}_n = 1)$ using the LIN-ESS algorithm, and they are evaluated on $m = 5000$ testing points.

Observation 1: Good classifiers are abundant. Our first observation is that, for reasonable n, most interpolating classifiers have good[1] test set performance. For example, for the MNIST dataset, we see that at $n = 350$, nearly 100% of the models that perfectly fit the training data achieve at least 95% ($\varepsilon = 0.05$) test accuracy. This indicates that, for this particular training set, bad classifiers (with error $> 5\%$) make up a set with very small measure. On the other hand, for the FASHION-MNIST task, only about 60% of classifiers perfectly fitting the training data get 95% test performance at $n = 350$ samples, but nearly 100% of such classifiers get 92% accuracy.

[1] Of course, one could fit a model from a more complicated function class and obtain even better test performance.

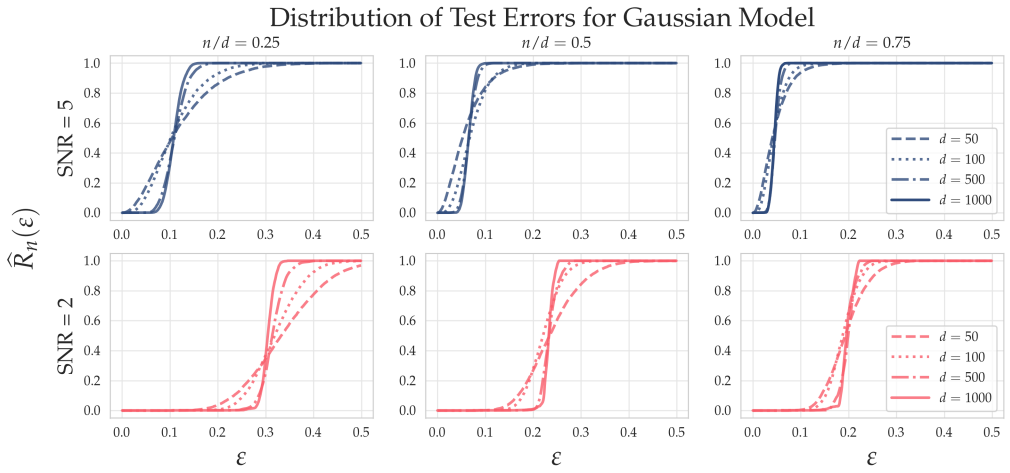

Figure 4: Plotting $\widehat{R}_n(\epsilon)$ for the Gaussian model (8) at various levels of d. **Blue** curves correspond to SNR = 5, **red** curves correspond to SNR = 2.

Observation 2: Existence of bad classifiers. A natural question that may arise out of these results is whether or not bad interpolating classifiers even exist for these tasks, at least for the parameter settings we consider. Here, we demonstrate a simple method for finding bad classifiers which, together with the previous results, shows that bad classifiers exist and constitute a tiny fraction of the version space. Given a dataset S_n, with $n < d$, we can append up to $n_b \leq (d-1)-n$ "bad" samples, to form a new dataset S_n' with $n' = n+n_b$ samples. Notice that any model $\mathbf{w} \in \mathsf{VS}(S_n')$ must also belong to $\mathsf{VS}(S_n)$, since $\mathsf{VS}(S_n') \subseteq \mathsf{VS}(S_n)$. Here, we construct $n_b = (d-1)-n$ "bad" points lying in the span of the set $\{-y_1\mathbf{x}_1, \ldots, -y_n\mathbf{x}_n\}$. In Figure 3, we plot the test error of interpolating classifiers constructed in this manner, fit using gradient descent with a logistic loss, for varying levels of n. We see that this method finds classifiers with test error that is nearly 1 for all values of n considered.

We are therefore left with an insightful contrast: in Figure 2, we observe that, for example, at $n = 350$, the set of interpolating MNIST classifiers with test accuracy $\geq 95\%$ comprise a set of measure essentially 1; while in Figure 3, we have demonstrated that there *exist* interpolating classifiers for this task with test accuracy nearly 0%. Thus, we see that the performance of the worst-case classifier gives basically no insight into the performance of the typical classifier, indicating that a uniform convergence-type analysis is not appropriate in this setting. This is also information that cannot be gleaned by looking at a summary statistic, like the *expected* test error of interpolating classifiers, i.e., $\mathbb{E}[\mathcal{E}_m(\mathbf{w}) \mid \mathsf{VS}(S_n)]$, alone—it is necessary to consider the full distribution.

3.2 Evaluation on Synthetic Datasets

For our next set of evaluations, we compute $R_n(\varepsilon)$ for synthetic data generated from the Gaussian mixture distribution

$$(\mathbf{x}, y) \sim \frac{1}{2}(N_+, 1) + \frac{1}{2}(N_-, -1)., \tag{8}$$

where $N_+ \sim \mathcal{N}(\mu, \Sigma)$, $N_- \sim \mathcal{N}(-\mu, \Sigma)$ and $\mu \in \mathbb{R}^d$, $\Sigma \in \mathcal{S}_+^d$. The purpose of this synthetic model is twofold. First, it allows us to demonstrate the ubiquity of the phenomena observed on the MNIST and FASHION-MNIST tasks. Second, it allows us to investigate the effect of varying the dimension d, which we could not do on the datasets studied in the previous section, as this was fixed at $d = 784$. This reveals that test errors begin to concentrate around a value ε^* as the dimension d increases.

For this model, we have that $y\mathbf{x} \sim \mathcal{N}(\mu, \Sigma)$, so we can characterize the set $\{\mathbf{w} : \mathcal{E}(\mathbf{w}) \leq \varepsilon\}$ with the condition

$$\mathcal{E}(\mathbf{w}) \leq \varepsilon \iff \frac{\mathbf{w}^\top \mu}{\sqrt{\mathbf{w}^\top \Sigma \mathbf{w}}} \geq -\Phi^{-1}(\varepsilon), \tag{9}$$

where $\Phi(\cdot)$ is the CDF of a $\mathcal{N}(0,1)$ distribution. Given a training set S_n and samples $\widehat{\mathbf{w}}_1, \ldots, \widehat{\mathbf{w}}_M \sim \mathbb{P}(\cdot \mid \mathsf{VS}(S_n))$, this expression allows us to compute an estimate $\widehat{R}_n(\varepsilon) = \frac{1}{M}\sum_{j=1}^M \mathbb{1}(\mathcal{E}(\widehat{\mathbf{w}}_j) \leq \varepsilon)$ in a straightforward manner.

As with many Gaussian models, the signal-to-noise ratio (SNR), which we define as $\sqrt{\mu^\top \Sigma^{-1} \mu}$ (or simply $\|\mu\|/\sigma$ when $\Sigma = \sigma^2 I$), controls much of the complexity of this task. In Figure 4, we plot $\widehat{R}_n(\varepsilon)$ for $d = 50, 100, 500, 1000$, and with SNR $= 2, 5$. For these experiments, we take $\Sigma = I$ and, to keep the

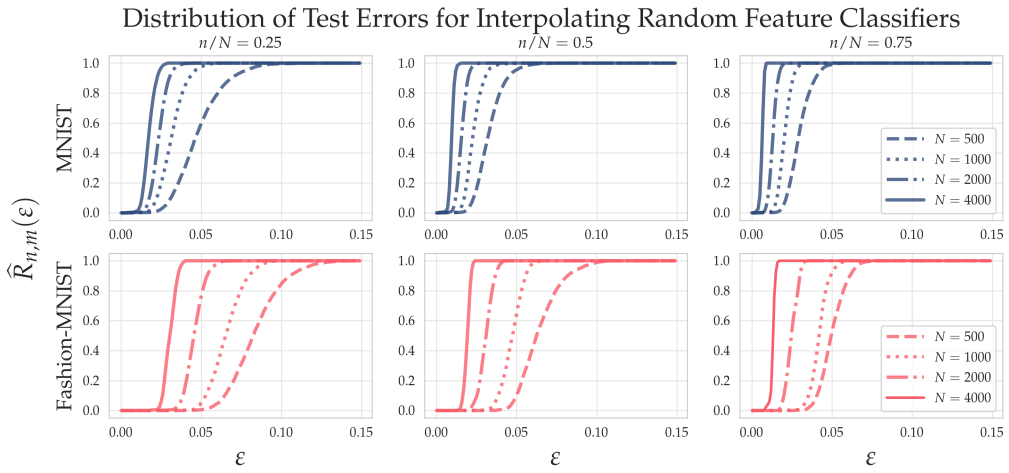

Figure 5: Plotting $\widehat{R}_{n,m}(\epsilon)$ for the random ReLU feature models on MNIST (0 vs 1) dataset (**blue**) and FASHION-MNIST (shirt vs pants) dataset (**red**).

SNR constant as we vary the dimension, we set $\mu = (\text{SNR}/\sqrt{d}, \ldots, \text{SNR}/\sqrt{d})^\top$.

Observation 3: Concentration at critical value ε^*. Our main observation here is the existence of a critical value ε^* around which test errors eventually concentrate. Indeed, we see in Figure 4 that as d grows, the distributions $R_n(\varepsilon)$ seem to approach the threshold function $\mathbb{1}(\varepsilon \geq \varepsilon^*)$ at a critical value ε^*, which depends on the aspect ratio $\alpha = n/d$. Therefore, in the large d regime, almost all interpolating classifiers have test error exactly ε^*, and so this critical value almost completely characterizes the distribution of test errors for interpolating classifiers. We also observe that this value is largely determined by the value of the SNR. In fact, we can derive a simple lower bound on the value of ε^*:

$$\varepsilon^* \geq \Phi(-\sqrt{\mu^\top \Sigma^{-1} \mu}). \tag{10}$$

This corresponds to the error of the optimal Bayes classifier $\mathbf{w}^\star = \Sigma^{-1}\mu$. In the next section, we observe a similar phenomenon for image classification tasks with random feature models.

4 RANDOM RELU FEATURES

In this section, we consider the class of random ReLU feature classifiers $\mathcal{F}_{\text{RRF}} = \{f(\mathbf{x}) = \text{sign}(\mathbf{w}^\top \phi(\mathbf{x})) : \mathbf{w} \in \mathbb{R}^N\}$, where $\phi(\mathbf{x}) = \sigma(\mathbf{U}\mathbf{x}) : \mathbb{R}^d \mapsto \mathbb{R}^N$. Here the rows $\mathbf{u}_1, \ldots, \mathbf{u}_N$ of $\mathbf{U}$ are drawn from the uniform distribution on the sphere $\mathbb{S}^{d-1}$ and $\sigma(z) = \max(z, 0)$ is the ReLU activation function. These can be viewed as one-layer ReLU networks with the weights of the first layer fixed, and they are known to enjoy universal approximation properties (Sun et al., 2019).

The benefit in studying such a model is that we can examine the behavior of the test error distributions as the number of hidden features N grows large, with $\alpha = n/N$ fixed. This allows us to observe the critical value behavior seen in linear classification with the Gaussian model (8), but this time with the image datasets MNIST and FASHION-MNIST.

In Figure 5, we plot the test error distributions for interpolating random ReLU classifiers on the MNIST and FASHION-MNIST tasks, for various number of hidden features N and ratios $\alpha = n/N$. Our main observation from these experiments is that, similar to the Gaussian model, as the number of features N grows, the test errors begin to concentrate around values $\varepsilon^* \equiv \varepsilon^*(\alpha)$. Like in the Gaussian model, the critical value depends on i) the difficulty of the task (it is larger for FASHION-MNIST than for MNIST) and ii) the aspect ratio $\alpha = n/N$. This finding indicates that the concentration phenomenon observed in Section 3.2 is quite general, and holds for both real and synthetic datasets.

We remark that the same technique used in Section 3.1 demonstrates that very poor classifiers also exist for the random ReLU classification models, and hence again verifies that the worst-case analysis of test errors is inappropriate for these models and datasets.

5 CHARACTERIZING THE DISTRIBUTION OF TEST ERRORS IN A SIMPLE MODEL

In this section, we present a simple model, and we prove that it exhibits the main qualitative properties we observed in Sections 3 and 4.

A full mathematical characterization of $R_{n,m}(\varepsilon)$ and/or $R_n(\varepsilon)$ is a challenging task. To see why,

let us define the random variables $\zeta_i = y_i \mathbf{w}^\top \phi(\mathbf{x}_i)$ for $(\mathbf{x}_i, y_i) \in S_n$ and $\zeta_{n+h} = y_{n+h}\mathbf{w}^\top\phi(\mathbf{x}_{n+h})$ for $(\mathbf{x}_{n+h}, y_{n+h}) \in S_{\text{test}}$ (where we emphasize that the randomness is due to $\mathbf{w}$). Then, for example, the normalization term $\mathbb{P}(\mathsf{VS}(S_n))$ can be expressed as

$$\begin{aligned}\mathbb{P}(\mathsf{VS}(S_n)) &= \int \prod_{i=1}^{n} \mathbb{1}(y_i \mathbf{w}^\top \phi(\mathbf{x}_i) \geq 0)\mathbb{P}(\mathrm{d}\mathbf{w}) \\ &= \mathbb{P}(\zeta_1 \geq 0, \zeta_2 \geq 0, \dots, \zeta_n \geq 0).\end{aligned} \tag{11}$$

That is, $\mathbb{P}(\mathsf{VS}(S_n))$ can be seen as an orthant probability under the distribution $\mathbb{P}$. When $\mathbb{P} = \mathcal{N}(\mathbf{0}, \mathbf{I})$, we find that $\zeta = (\zeta_1, \zeta_2, \dots, \zeta_n) \sim \mathcal{N}(\mathbf{0}, \mathbf{A}\mathbf{A}^\top)$, where $\mathbf{A}$ is the $n \times N$ matrix whose i^{th} row is $(y_i\phi(\mathbf{x}_i))^\top$ and whose $(i,j)^{\text{th}}$ entry is $y_i y_j \phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j)$. Computing such a Gaussian orthant probability for a general covariance matrix is a classical problem, and explicit formulae for them are known only in dimensions ≤ 5 and in a few other special cases (Dunnett and Sobel, 1955, Steck, 1962, Abrahamson, 1964).

Hence, to present a model we can analyze, here we consider a simplified setting where the testing and training samples have a fixed positive correlation with each other, i.e., for fixed $\rho \in (0, 1]$,

$$(\mathbf{A}\mathbf{A}^\top)_{ij} = y_i y_j \phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j) = \rho, \tag{12}$$

for each pair of indices $i \neq j$ in $S_n \cup S_{\text{test}}$ (where here we assume $\phi(\mathbf{x}_i)$ are normalized to have unit ℓ^2 norm, without loss of generality).[2] Under this assumption, we can leverage implicit expressions for the normalizing term $\mathbb{P}(\mathsf{VS}(S_n))$, which makes the problem more amenable to analysis.

We remark that to derive asymptotically valid expressions for $R_n(\varepsilon)$ and $R_{n,m}(\varepsilon)$, one may be tempted to approximate (11) using off-the-shelf techniques for approximating high-dimensional integrals, e.g., Laplace's method. However, there are a number of pitfalls with this approach. First, it is difficult to quantify the approximation errors, and results that do exist are not precise enough for our purposes. Second, certain conditions for Laplace's method or other standard integral expansions do not hold in our setting.[3] Nevertheless, we can leverage special properties of the Gaussian distribution and quantile functions to prove several nontrivial results. Henceforth, for sequences $\{a_n\}$ and $\{b_n\}$, the notation $a_n \sim b_n$ means $a_n = b_n(1 + o(1))$ as $n \to \infty$.[4]

Our first result considers the setting of a single testing point $(\mathbf{x}_{n+1}, y_{n+1})$, and it demonstrates the effect of a larger correlation ρ on the probability of correctly classifying a new test point. Furthermore, it shows that, at least for this simple setting, we can expect the probability of correctly classifying a testing point to converge to 1 at a $O(1/n)$ rate.

Theorem 1. *Suppose we have a single testing point $(\mathbf{x}_{n+1}, y_{n+1})$, which together with the training data satisfies the correlation structure* (12)*. Then, as $n\rho \to \infty$,*

$$\mathbb{P}(y_{n+1} = \operatorname{sign}(\mathbf{w}^\top \phi(\mathbf{x}_{n+1})) \mid \mathsf{VS}(S_n)) \sim 1 - \frac{1-\rho}{n\rho}. \tag{13}$$

The proof of Theorem 1 relies mainly on a new asymptotic formula for the orthant probability of equicorrelated Gaussian random variables. To the best of our knowledge, this is the first of its kind, and it may be of independent interest. We state this result below in the following Lemma.

Lemma 1. *Let $\rho \in [0, 1)$ and $(X_1, ..., X_n) \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ with $\mathbf{\Sigma}_{ij} = \rho$ for $i \neq j$ and $\Sigma_{ii} = 1$ for all i. Then as $n\rho \to \infty$,*

$$\begin{aligned}&\mathbb{P}(X_1 \geq 0, X_2 \geq 0, \dots, X_n \geq 0) \sim \\ &\qquad \sqrt{\frac{1-\rho}{\rho}}\Gamma\left(\frac{1-\rho}{\rho}\right)(4\pi \log(n))^{\frac{1}{2}(\frac{1-\rho}{\rho}-1)} n^{-\frac{1-\rho}{\rho}}.\end{aligned}$$

Theorem 1 then follows by carefully evaluating the ratio of the above expression at $n+1$ and n.

Before stating our next result, we provide a formal definition of a critical value ε^* which we will reference therein.

Definition 2. *We say that ε^* is a critical value if, for each $c > 0$, $R_n(\varepsilon^* - c) = 0$ and $R_n(\varepsilon^* + c) \to 1$ as $n \to \infty$.*

Our next result provides a connection between the critical value ε^*, the number of training samples, and the correlation ρ.

Theorem 2. *Suppose the testing and training data satisfies the correlation structure* (12)*. Let U be a gamma random variable with shape and scale parameters $(1-\rho)/\rho$ and 1, respectively, i.e., $U \sim \text{Gamma}(\frac{1-\rho}{\rho}, 1)$. Then, as $n\rho \to \infty$,*

$$R_n(\varepsilon) \sim \mathbb{P}(U \leq n\varepsilon). \tag{14}$$

In particular, as $n\rho \to \infty$,

$$\varepsilon^* = \frac{1-\rho}{n\rho} \tag{15}$$

is a critical value.

In this simple setting, n and ρ completely determine the distribution $R_n(\varepsilon)$: if ρ is close to 1, then the

[2] By correlation between data points, we mean $y_i y_j \phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j)$ for $i \neq j$.

[3] For example, the maximum of the function in the exponent of the integrand occurs at infinity.

[4] That is, it should not be confused with "has the probability distribution of" which uses the same notation.

data points are nearly parallel, and we will have that the test errors sharply concentrate around the critical value ε^*, even for n small. Of course, in practice, there will be a more subtle and complicated relationship between the correlations and the full distribution $R_n(\varepsilon)$, which will likely be difficult to characterize precisely. Nonetheless, we believe that it may be possible to prove concentration in the general case, without explicitly characterizing the full distribution $R_n(\varepsilon)$. This is captured by the following conjecture.

Conjecture 1. *For any model class $\mathcal{F}_\phi$, datasets S_n, testing distribution $\Pr_{\mathbf{x},y}$ (each potentially satisfying some regularity conditions) and scaling $0 < \alpha < 1$, there exists a critical value $\varepsilon^*(\alpha)$ such that $\lim_{n,N\to\infty, n/N\to\alpha} R_n(\varepsilon) = \mathbb{1}(\varepsilon \geq \varepsilon^*(\alpha))$ almost surely.*

Theorem 2 provides such a result in the case when the data is equicorrelated. Previous work using the statistical mechanics framework also prove similar results under different simplifying assumptions, namely when the features $\mathbf{x}_{ik} \overset{\text{i.i.d.}}{\sim} \text{Unif}(\{-1, 1\})$, $k = 1, \ldots, d$, and the labels y_i are generated via a teacher model $\mathbf{w}_\star$ s.t. $y_i = \text{sign}(\mathbf{w}_\star^\intercal \mathbf{x}_i)$ (see, e.g., Chapter 2 of (Engel and Van den Broeck, 2001)). However, these results typically only focus on the $n > d$ case, which is less relevant to the modern machine learning regime.

6 DISCUSSION AND CONCLUSION

In this paper, we built on previous literature on the statistical mechanics of learning to develop a framework to study the *typical* test error of a classifier, and we propose this as an alternative to the more standard uniform convergence approach. We formally define the full distribution of test errors among interpolating classifiers and introduce a method to compute this distribution accurately on real datasets. One of the most important findings of our investigation is that, given a particular training and testing setup, there exists a critical value ε^* around which almost all interpolating classifiers' test errors eventually concentrate. This will not come as a surprise to the statistical physicist: such typical values commonly appear in physical systems. However, as we have demonstrated, this critical value can differ significantly from the error $\varepsilon_{\text{unif}}$, which one would obtain via a uniform convergence analysis, especially in the interpolating/over-parameterized regime, and which may be more familiar to the machine learner.

Our results should motivate further research into alternatives to the uniform convergence framework, either through the lens of statistical physics or some other (likely related) perspective, and they should ultimately help resolve questions surrounding the good performance of over-parameterized machine learning models. As a first step, we state a few potential directions for future work building off of the results presented here.

More general function classes. While encompassing many models of interest, the function classes $\mathcal{F}_\phi$ of course do not include general neural network architectures. In this paper, we studied random feature models, which can be interpreted as neural networks with internal weights fixed at a random initialization. Another interesting setting which may be more tractable to study would be that of linearized networks of the form

$$f(\mathbf{x}) = \text{sign}(\mathbf{w}^\intercal \nabla F(\mathbf{x}; \mathbf{w}_0)) \tag{16}$$

where F is an arbitrary neural network with random initialization $\mathbf{w}_0$. A variety of results have shown that these models coincide with neural networks in the large-width limit via the neural tangent kernel (Jacot et al., 2018, Arora et al., 2019). While our approach would, in theory, work out-of-the-box for these models, in practice, these involve a very large number of features (approximately $O(LN^2)$, where L is the number of layers, and N is the width of each layer). We found that even with the LIN-ESS algorithm, sampling from $\mathbb{P}(\cdot \mid \mathsf{VS}(S_n))$ was impractical for these models. However, developing other methods for computation in this setting could yield interesting insights into the advantages (and disadvantages) of various network architectures.

Beyond the interpolating regime. The motivation for our studying interpolating classifiers comprising the version space $\mathsf{VS}(S_n)$ was previous work in the statistical mechanics literature, as well as the well-known worst-case results for these models given by, e.g., Vapnik–Chervonenkis theory. However, this is not the only method one could use to study the distribution of test errors. A promising alternative would be to consider the distribution over weights $\mathbf{w}$ induced by some optimization algorithm, such as stochastic gradient descent (SGD). Indeed, previous work has shown that under various assumptions, SGD produces a Gaussian stationary distribution over weights $\mathbf{w}$ (Mandt et al., 2017). Under other (probably more realistic) assumptions, it leads to heavy-tailed structure in the weights (Hodgkinson and Mahoney, 2020, Gurbuzbalaban et al., 2020). An intriguing direction for future work would be to study the distribution over test errors $\mathcal{E}(\mathbf{w})$ induced by such a stationary distribution. It is possible that this may even simplify the theoretical investigation: whereas we studied weights drawn from $\mathbb{P}(\cdot \mid \mathsf{VS}(S_n))$ (a rather complicated distribution), it may be easier to study weights drawn from a Gaussian (or some other tractable) distribution.

Acknowledgments

MM would like to acknowledge DARPA, NSF, and ONR for providing partial support of this work. JK would like to acknowledge funding from NSF DMS-1915932 and TRIPODS DATA-INSPIRE CCF-1934924. We also thank the authors of (Gessner et al., 2020) for sharing their implementation of the LIN-ESS algorithm.

References

[Abrahamson, 1964] Abrahamson, I. G. (1964). Orthant Probabilities for the Quadrivariate Normal Distribution. *Annals of Mathematical Statistics*, 35(4):1685–1703.

[Arora et al., 2019] Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R., and Wang, R. (2019). On Exact Computation with an Infinitely Wide Neural Net. In *Advances in Neural Information Processing Systems*.

[Barbier et al., 2019] Barbier, J., Krzakala, F., Macris, N., Miolane, L., and Zdeborová, L. (2019). Optimal errors and phase transitions in high-dimensional generalized linear models. *Proceedings of the National Academy of Sciences*, 116(12):5451–5460.

[Bartlett et al., 2020] Bartlett, P. L., Long, P. M., Lugosi, G., and Tsigler, A. (2020). Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070.

[Belkin et al., 2019] Belkin, M., Hsu, D., Ma, S., and Mandal, S. (2019). Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences of the United States of America*, 116(32):15849–15854.

[Candes and Sur, 2018] Candes, E. J. and Sur, P. (2018). The phase transition for the existence of the maximum likelihood estimate in high-dimensional logistic regression. Technical Report arXiv preprint: 1804.09753.

[Chatterji and Long, 2020] Chatterji, N. S. and Long, P. M. (2020). Finite-sample analysis of interpolating linear classifiers in the overparameterized regime. Technical Report arXiv preprint: 2004.12019.

[Choromanska et al., 2015] Choromanska, A., Henaff, M., Mathieu, M., Arous, G. B., and LeCun, Y. (2015). The Loss Surfaces of Multilayer Networks. In *18th International Conference on Artificial Intelligence and Statistics*.

[Deng et al., 2020] Deng, Z., Kammoun, A., and Thrampoulidis, C. (2020). A model of double descent for high-dimensional logistic regression. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4267–4271.

[Dereziński et al., 2019] Dereziński, M., Liang, F., and Mahoney, M. W. (2019). Exact expressions for double descent and implicit regularization via surrogate random design. Technical Report arXiv preprint: 1912.04533.

[Dunnett and Sobel, 1955] Dunnett, C. W. and Sobel, M. (1955). Approximations to the Probability Integral and Certain Percentage Points of a Multivariate Analogue of Student's t-Distribution. *Biometrika*, 42(1/2):258.

[Engel and Van den Broeck, 2001] Engel, A. and Van den Broeck, C. (2001). *Statistical Mechanics of Learning*. Cambridge University Press.

[Gessner et al., 2020] Gessner, A., Kanjilal, O., and Hennig, P. (2020). Integrals over Gaussians under Linear Domain Constraints. In *Proceedings of Machine Learning Research*.

[Gunasekar et al., 2018] Gunasekar, S., Woodworth, B., Bhojanapalli, S., Neyshabur, B., and Srebro, N. (2018). Implicit regularization in matrix factorization. In *2018 Information Theory and Applications Workshop, ITA 2018*. Institute of Electrical and Electronics Engineers Inc.

[Gurbuzbalaban et al., 2020] Gurbuzbalaban, M., Simsekli, U., and Zhu, L. (2020). The heavy-tail phenomenon in SGD. Technical Report Preprint: arXiv:2006.04740.

[Hastie et al., 2019] Hastie, T., Montanari, A., Rosset, S., and Tibshirani, R. J. (2019). Surprises in High-Dimensional Ridgeless Least Squares Interpolation. Technical Report arXiv preprint: 1903.08560.

[Haussler et al., 1996] Haussler, D., Kearns, M., Sebastian Seung, H., and Tishby, N. (1996). Rigorous learning curve bounds from statistical mechanics. *Machine Learning*, 25(2-3):195–236.

[Hodgkinson and Mahoney, 2020] Hodgkinson, L. and Mahoney, M. W. (2020). Multiplicative noise and heavy tails in stochastic optimization,. Technical Report Preprint: arXiv:2006.06293.

[Jacot et al., 2018] Jacot, A., Gabriel, F., and Hongler, C. (2018). Neural Tangent Kernel: Convergence and Generalization in Neural Networks.

In *Advances in Neural Information Processing Systems*.

[Liao et al., 2020] Liao, Z., Couillet, R., and Mahoney, M. W. (2020). A random matrix analysis of random Fourier features: beyond the Gaussian kernel, a precise phase transition, and the corresponding double descent. Technical report. arXiv preprint: 2006.05013.

[Ma et al., 2020] Ma, C., Wang, K., Chi, Y., and Chen, Y. (2020). Implicit Regularization in Nonconvex Statistical Estimation: Gradient Descent Converges Linearly for Phase Retrieval, Matrix Completion, and Blind Deconvolution. *Foundations of Computational Mathematics*, 20(3):451–632.

[Mahoney, 2012] Mahoney, M. W. (2012). Approximate computation and implicit regularization for very large-scale data analysis. In *Proceedings of the 31st ACM Symposium on Principles of Database Systems*, pages 143–154.

[Mandt et al., 2017] Mandt, S., Hoffman, M. D., and Blei, D. M. (2017). Stochastic Gradient Descent as Approximate Bayesian Inference. *Journal of Machine Learning Research*, 18:1–35.

[Martin and Mahoney, 2017] Martin, C. H. and Mahoney, M. W. (2017). Rethinking generalization requires revisiting old ideas: statistical mechanics approaches and complex learning behavior. Technical Report arXiv preprint:1710.09553.

[Murray et al., 2010] Murray, I., Prescott, R., David, A., and Mackay, J. C. (2010). Elliptical slice sampling. In *13th International Conference on Artificial Intelligence and Statistics*.

[Nagarajan and Kolter, 2019] Nagarajan, V. and Kolter, J. Z. (2019). Uniform convergence may be unable to explain generalization in deep learning. In *Advances in Neural Information Processing Systems*.

[Opper and Haussler, 1991] Opper, M. and Haussler, D. (1991). Calculation of the Learning Curve of Bayes Optimal Classiication Algorithm for Learning a Perceptron With Noise. In *Proceedings of the Fourth Annual Workshop on Computational Learning Theory*, pages 75–87.

[Seung et al., 1992] Seung, H. S., Sompolinsky, H., and Tishby, N. (1992). Statistical mechanics of learning from examples. *Physical Review A*, 45(8):6056–6091.

[Soudry et al., 2018] Soudry, D., Hoffer, E., and Srebro, N. (2018). The implicit bias of gradient descent on separable data. In *International Conference on Learning Representations*.

[Steck, 1962] Steck, G. (1962). Orthant Probabilities for the Equicorrelated Multivariate Normal Distribution. *Biometrika*, 49(3/4):433–445.

[Sun et al., 2019] Sun, Y., Gilbert, A., and Tewari, A. (2019). On the Approximation Properties of Random ReLU Features. Technical Report arXiv preprint: 1810.04374.

[Theisen et al., 2020] Theisen, R., Klusowski, J. M., and Mahoney, M. W. (2020). Good classifiers are abundant in the interpolating regime. Technical report. arXiv preprint: 2006.12625.

[Watkin et al., 1993] Watkin, T. L. H., Rau, A., and Biehl, M. (1993). The statistical mechanics of learning a rule. *Rev. Mod. Phys.*, 65(2):499–556.

[Wu and Zhu, 2017] Wu, L. and Zhu, Z. (2017). Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes. In *ICML 2017 Workshop on Principled Approaches to Deep Learning*.

[Xing et al., 2019] Xing, Y., Song, Q., and Cheng, G. (2019). Benefit of Interpolation in Nearest Neighbor Algorithms. Technical Report arXiv preprint: 1909.11720.