On the Role of Optimization in Double Descent: A Least Squares Study

Anonymous Author(s)
Affiliation
Address
email

Abstract

Empirically it has been observed that the performance of deep neural networks steadily improves as we increase model size, contradicting the classical view on overfitting and generalization. Recently, the *double descent* phenomena has been proposed to reconcile this observation with theory, suggesting that the test error has a second descent when the model becomes sufficiently overparametrized, as the model size itself acts as an implicit regularizer. In this paper we add to the growing body of work in this space, providing a careful study of learning dynamics as a function of model size for the least squares scenario. We show an excess risk bound for the gradient descent solution of the least squares objective. The bound depends on the smallest non-zero eigenvalue of the covariance matrix of the input features, via a functional form that has the double descent behaviour. This gives a new perspective on the double descent curves reported in the literature. Our analysis of the excess risk allows to decouple the effect of optimisation and generalisation error. In particular, we find that in case of noiseless regression, double descent is explained solely by optimisation-related quantities, which was missed in studies focusing on the Moore-Penrose pseudoinverse solution. We believe that our derivation provides an alternative view compared to existing work, shedding some light on a possible cause of this phenomena, at least in the considered least squares setting. We empirically explore if our predictions hold for neural networks, in particular whether the covariance of intermediary hidden activations has a similar behaviour as the one predicted by our derivations.

1 Introduction

Deep Neural Networks have shown amazing versatility across a large range of domains. Among one of their main features is their ability to perform better with scale. Indeed, some of the most impressive results [see e.g. Brock et al., 2021, Brown et al., 2020, Senior et al., 2020, Schrittwieser et al., 2020, Silver et al., 2017, He et al., 2016 and references therein] have been obtained often by exploiting this fact, leading to models that have at least as many parameters as the number of examples in the dataset they are trained on. Empirically, the limitation on the model size seems to be mostly imposed by hardware or compute. From a theoretical point of view, however, this property is quite surprising and counter-intuitive, as one would expect that in such extremely overparametrized regimes the learning would be prone to overfitting [Hastie et al., 2009, Shalev-Shwartz and Ben-David, 2014].

Recently Belkin et al. [2019] proposed Double Descent (DD) phenomena as an explanation. They argue that the classical view of overfitting does not apply in extremely over-parameterized regimes, which were less studied prior to the emergence of the deep learning era. The classical view in the parametric learning models was based on error curves showing that the training error decreases monotonically when plotted against model size, while the corresponding test errors displayed a

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

U-shape curve, where the model size for the bottom of the U-shape was taken to achieve the ideal trade-off between model size and generalization, and larger model sizes than that were thought to lead to 'overfitting' since the gap between test errors and training errors increased.

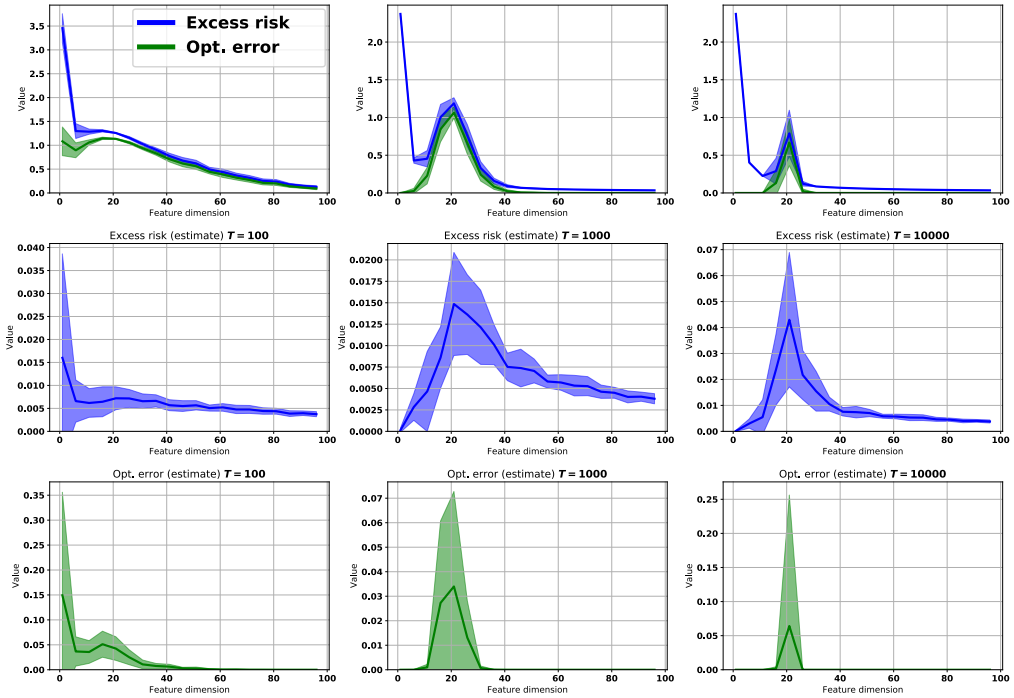

Figure 1: Evaluation of a synthetic setting inspired by Belkin et al. [2020]. We consider a linear regression problem ($n = 20$, $d \in [100]$), where regression parameters are fixed, and instances are sampled from $[-1, 1]$-truncated normal density. GD is run with $\alpha = 0.05$ and initialization variance is set as $\nu_{\text{init}}^2 = 1/d$. The first row demonstrates behavior of (1), the second shows an estimate of the excess risk (on 10^4 held-out points), and the third an estimate of the optimization error.

The classical U-shape error curve dwells in what is now called the under-parametrized regime, where the model size is smaller than the size of the dataset. Arguably, the restricted model sizes used in the past were tied to the available computing power. By contrast, it is common nowadays for model sizes to be larger than the amount of available data, which we call the over-parametrized regime. The divide between these two regimes is marked by a point where model size matches dataset size, which Belkin et al. [2019] called the *interpolation threshold*.

The work of Belkin et al. [2019] argues that as model size grows beyond the interpolation threshold, one will observe a second descent of the test error that asymptotes in the limit to smaller values than those in the underparametrized regime, which indicates better generalization rather than overfitting. To some extent this was already known in the *nonparametric* learning where model complexity scales with the amount of data by design (such as in nearest neighbor rules and kernels), yet one can generalize well and even achieve statistical consistency [Györfi et al., 2002]. This has lead to a growing body of works trying to identify the mechanisms behind DD, to which the current manuscript belongs too. We refer the reader to Appendix A, where the related literature is discussed. Similar to these works, our goal is also to understand the cause of DD. Our approach is slightly different: we explore the least squares problem that allows us to work with analytic expressions for all the quantities involved. Fig. 1 provides a summary of our findings. In particular, it shows the behaviour of the excess risk in a setting with random inputs and noise-free labels, for which in Section 2 we prove a bound that has the form $\mathbb{E}\left[(1 - \alpha\widehat{\lambda}_{\min}^{+})^{2T}\right]\|\boldsymbol{w}^{\star}\|^2 + \frac{\|\boldsymbol{w}^{\star}\|^2}{\sqrt{n}}$. In this setting, the linear predictors project d-dimensional features by dot product with a weight vector which must be learned from data; then $\boldsymbol{w}^{\star}$ refers to the optimal solution, α is a constant learning rate, and n is the number of examples in the training set. Note that the feature dimension d coincides with the number of parameters in this particular setting, hence $d > n$ is the overparametrized regime. The quantity $\widehat{\lambda}_{\min}^{+}$ is of special

importance: *It is the smallest positive eigenvalue of the sample covariance matrix of the features.* In particular, we observe that the excess risk is controlled by the smallest non-zero eigenvalue of the covariance of the features, and its functional dependence exhibits a profile similar to the DD curve. This offers a new perspective on the problem.

In Fig. 1 we observe *a peaking behavior*, not only in the excess risk, but also in the quantity that we label 'optimization error' which is a special term of the excess risk bound that is purely related to optimization. The peaking behaviour of the excess risk (MSE in case of the square loss) was observed and studied in a number of settings [Belkin et al., 2019, Mei and Montanari, 2019, Derezinski et al., 2020]; however, the connection between the peaking behavior and optimization so far received less attention. This pinpoints a less-studied setting and we conjecture that the DD phenomenon occurs due to $\widehat{\lambda}^{+}_{\min}$. In the absence of label noise, we conclude that DD manifests due to the optimization process. On the other hand, when label noise is present, in addition to the optimization effect, $\widehat{\lambda}^{+}_{\min}$ also has an effect on the generalization error.

Our contributions: Our main theoretical contribution is provided in Section 2. In particular, Section 2.1 focuses on the noise-free least squares problem, Section 2.2 adds noise to the problem, and Section 2.3 deals with concentration of the sample-dependent $\widehat{\lambda}^{+}_{\min}$ around its population counterpart. Sections 3 and 4 provide an in-depth discussion on the implications of our findings and an empirical exploration of the question whether simple neural networks have a similar behaviour.

Notation: The linear algebra/analysis notation used in this work is defined in Appendix B. We briefly mention here that we denote column vectors and matrices with small and capital bold letters, respectively, e.g. $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_d]^\top \in \mathbb{R}^d$ and $\boldsymbol{A} \in \mathbb{R}^{d_1 \times d_2}$. Singular values of a rectangular matrix $\boldsymbol{A} \in \mathbb{R}^{n \times d}$ are denoted by $s_{\max}(\boldsymbol{A}) = s_1(\boldsymbol{A}) \geq \ldots \geq s_{n \wedge d}(\boldsymbol{A}) = s_{\min}(\boldsymbol{A})$. The rank of $\boldsymbol{A}$ is $r = \max\{k \mid s_k(\boldsymbol{A}) > 0\}$. Eigenvalues of a Positive Semi-Definite (PSD) matrix $\boldsymbol{M} \in \mathbb{R}^{d \times d}$ are nonnegative and are denoted $\lambda_{\max}(\boldsymbol{M}) = \lambda_1(\boldsymbol{M}) \geq \ldots \geq \lambda_d(\boldsymbol{M}) = \lambda_{\min}(\boldsymbol{M})$, while the smallest *non-zero* eigenvalue is denoted $\lambda^{+}_{\min}(\boldsymbol{M})$.

Next, we set the learning theory notation. In a parametric statistical learning problem the learner is given a training set $S = (Z_1, \ldots, Z_n)$, which is an n-tuple consisting of independent random elements, called training examples, distributed according to some unknown distribution $\mathcal{D} \in \mathcal{M}_1(\mathcal{Z})$, where $\mathcal{Z}$ is called the *example space*. The learner's goal is to select parameter $\boldsymbol{w}$ from some *parameter space* $\mathcal{W}$ so as to minimize the *population loss* $L(\boldsymbol{w}) = \int_{\mathcal{Z}} \ell(\boldsymbol{w}, z)\mathcal{D}(\mathrm{d}z)$, where $\ell : \mathcal{W} \times \mathcal{Z} \to [0, 1]$ is some given *loss function*. A learner following the Empirical Risk Minimization (ERM) principle selects a $\boldsymbol{w}$ with the smallest *empirical loss* $\hat{L}_S(\boldsymbol{w}) = (\ell(\boldsymbol{w}, Z_1) + \cdots + \ell(\boldsymbol{w}, Z_n))/n$ over the training set. In this report we consider a Euclidean parameter space: $\mathcal{W} = \mathbb{R}^d$.

We consider a least squares regression problem. In this setting, each example is an instance-label pair: $Z_i = (\boldsymbol{X}_i, Y_i) \in \mathcal{B}_1 \times [0, 1]$. We assume that inputs $\boldsymbol{X}_i$ are from the Euclidean ball of unit radius $\mathcal{B}_1 \subset \mathbb{R}^d$, and labels Y_i are in the unit interval $[0, 1]$. For a suitably chosen parameter vector $\boldsymbol{w}$, the noiseless regression model is $f(\boldsymbol{X}) = \boldsymbol{X}^\top \boldsymbol{w}$ and the model with label noise is $f(\boldsymbol{X}) = \boldsymbol{X}^\top \boldsymbol{w} + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma^2)$. The loss function is the square loss: $\ell(\boldsymbol{w}, Z_i) = (f(\boldsymbol{X}_i) - Y_i)^2/2$.

2 Excess Risk of the Gradient Descent Solution

We focus on learners that optimize parameters via the Gradient Descent algorithm. We treat GD as a measurable map $\mathcal{A} : \mathcal{S} \times \mathbb{R}^d \to \mathbb{R}^d$, where $\mathcal{S} = \mathcal{Z}^n$ is the space of size-n training sets. Given a training set $S \in \mathcal{S}$ and an initialization point $\boldsymbol{w}_0 \in \mathcal{W}$, we write $\mathcal{A}_S(\boldsymbol{w}_0)$ to indicate the output obtained recursively by the standard GD update rule with some fixed step size $\alpha > 0$, i.e. $\mathcal{A}_S(\boldsymbol{w}_0) = \boldsymbol{w}_T$, where

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} - \alpha \nabla \widehat{L}_S(\boldsymbol{w}_{t-1}), \qquad t = 1, \ldots, T\,.$$

We look at the behavior of GD in the *overparametrized* regime ($d > n$) when the initialization parameters are sampled from an isotropic Gaussian density, that is $\boldsymbol{W}_0 \sim \mathcal{N}(\boldsymbol{0}, \nu^2_{\text{init}} \boldsymbol{I}_{d \times d})$ with some initialization variance ν^2_{init}. It is well-known that in the overparametrized regime, GD is able to achieve zero empirical loss. Therefore, rather than focusing on the generalization gap

$L(\mathcal{A}_S(\boldsymbol{W}_0)) - \hat{L}_S(\mathcal{A}_S(\boldsymbol{W}_0))$ it is natural to compare the loss of $\mathcal{A}_S(\boldsymbol{W}_0)$ to that of the *best* possible predictor. Thus, we consider the *excess risk* defined as

$$\mathcal{E}(\boldsymbol{w}^\star) = L(\mathcal{A}_S(\boldsymbol{W}_0)) - L(\boldsymbol{w}^\star) \ , \qquad \boldsymbol{w}^\star \in \arg\min_{\boldsymbol{w}\in\mathbb{R}^d} L(\boldsymbol{w}) \ .$$

Our results are based on a the requirement that $\mathcal{A}_S$ satisfies the following regularity condition:

Definition 1. *A map $f : \mathbb{R}^d \to \mathbb{R}^d$ is called $(\Delta, \boldsymbol{M})$-admissible, where $\boldsymbol{M}$ is a fixed PSD matrix and $\Delta \geq 0$, if for all $\boldsymbol{w}, \boldsymbol{w}' \in \mathbb{R}^d$ the following holds:*

$$\|f(\boldsymbol{w}) - f(\boldsymbol{w}')\|_{\boldsymbol{M}} \leq \Delta \|\boldsymbol{w} - \boldsymbol{w}'\| \ .$$

Notice that the norm on the left-hand side is $\|\cdot\|_{\boldsymbol{M}}$, while that on the right-hand side is the standard Euclidean norm. Also note that this inequality entails a Lipschitz condition with Lipschitz factor Δ.

Our first main result gives an upper bound on the excess risk of GD output, assuming that the output of $\mathcal{A}_S$ is of *low-rank*, in the sense that for some low-rank orthogonal projection $\boldsymbol{M} \in \mathbb{R}^{d\times d}$ we assume that $\boldsymbol{M}\mathcal{A}_S(\boldsymbol{w}) = \mathcal{A}_S(\boldsymbol{w})$ almost surely (a.s.) with respect to S, for any initialization $\boldsymbol{w}$. This condition is of interest in the overparameterized regime, where the learning dynamics effectively happens in a subspace which is arguably of much smaller dimension than the whole parameter space. The following theorem bounds the excess risk (with respect to a possibly non-convex but smooth loss) of any algorithm that satisfies Definition 1 with some $(\Delta, \boldsymbol{M})$. Later it will become apparent that in a particular learning problem this pair consists of data-dependent quantities. Importantly, the theorem demonstrates how the excess risk is controlled by the learning dynamics on the subspace spanned by $\boldsymbol{M}$ (the first and the second terms on the right hand side). It also shows how much is lost due to not learning on the complementary subspace (the third term). The first two terms will become crucial in our analysis of the double descent, while we will show that the last term will vanish as $n \to \infty$.

Theorem 1 (Excess Risk)**.** *Assume that $\boldsymbol{W}_0 \sim \mathcal{N}(\boldsymbol{0}, \nu_{\text{init}}^2 \boldsymbol{I}_{d\times d})$, and assume that $\mathcal{A}_S$ is $(\Delta, \boldsymbol{M})$-admissible (Definition 1), where Δ and $\boldsymbol{W}_0$ are independent. Further assume $\boldsymbol{M}\mathcal{A}_S(\boldsymbol{w}) = \mathcal{A}_S(\boldsymbol{w})$ for any $\boldsymbol{w}$, and that L and $\hat{L}$ are H-smooth. Then, for any $\boldsymbol{w}^\star \in \arg\min_{\boldsymbol{w}\in\mathbb{R}^d} L(\boldsymbol{w})$ we have*

$$\mathbb{E}[\mathcal{E}(\boldsymbol{w}^\star)] \leq H \left(\underbrace{\mathbb{E}[\Delta^2]\left(\|\boldsymbol{w}^\star\|^2 + \nu_{\text{init}}^2(2+d)\right)}_{(1)} + \underbrace{\mathbb{E}[\|\mathcal{A}_S(\boldsymbol{w}^\star) - \boldsymbol{w}^\star\|_{\boldsymbol{M}}^2]}_{(2)} + \frac{1}{2}\underbrace{\mathbb{E}[\|\boldsymbol{w}^\star\|_{\boldsymbol{I}-\boldsymbol{M}}^2]}_{(3)} \right) \ .$$

In particular for GD, having $\alpha \leq 1/H$,

$$\mathbb{E}[\|\mathcal{A}_S(\boldsymbol{w}^\star) - \boldsymbol{w}^\star\|_{\boldsymbol{M}}^2] \leq 2\alpha T L(\boldsymbol{w}^\star) \ .$$

The proof is in Appendix D. The main steps are using the H-smoothness of L to upper-bound $\mathcal{E}(\boldsymbol{w}^\star)$ in terms of the squared norm of $\mathcal{A}_S(\boldsymbol{W}_0) - \boldsymbol{w}^\star$ and decomposing the latter as the sum of the squared norms of its projections onto the space spanned by M and its orthogonal complement, by the Pythagorean theorem. Then $\mathcal{A}_S(\boldsymbol{W}_0) - \boldsymbol{w}^\star = \mathcal{A}_S(\boldsymbol{W}_0) - \mathcal{A}_S(\boldsymbol{w}^\star) + \mathcal{A}_S(\boldsymbol{w}^\star) - \boldsymbol{w}^\star$ is used on the subspace spanned by $\boldsymbol{M}$: the norm of $\mathcal{A}_S(\boldsymbol{W}_0) - \mathcal{A}_S(\boldsymbol{w}^\star)$ is controlled by using the admissibility of $\mathcal{A}_S$ and Gaussian integration, and the norm of $\mathcal{A}_S(\boldsymbol{w}^\star) - \boldsymbol{w}^\star$ is controlled by the accumulated squared norms of gradients of $\hat{L}_S$ over T steps of gradient descent, which is conveniently bounded by $2\alpha T \hat{L}_S(\boldsymbol{w}^\star)$ when $\alpha \leq 1/H$ due to the H-smoothness of $\hat{L}_S$.

We will rely on Theorem 1 for our analysis of the Least-Squares problem as follows.

2.1 Least-Squares with Random Design and No Label Noise

Consider a noise-free linear regression model with random design:

$$Y = \boldsymbol{X}^\top \boldsymbol{w}^\star$$

where instances $\boldsymbol{X}$ are distributed according to some unknown distribution P_X supported on a d-dimensional unit Euclidean ball. After observing a training sample $S = ((\boldsymbol{X}_i, Y_i))_{i=1}^n$, we run GD on the given empirical square loss

$$\hat{L}_S(\boldsymbol{w}) = \frac{1}{2n}\sum_{i=1}^n (\boldsymbol{w}^\top \boldsymbol{X}_i - Y_i)^2 \ .$$

In the setting of our interest, the sample covariance matrix $\widehat{\boldsymbol{\Sigma}} = (\boldsymbol{X}_1\boldsymbol{X}_1^\top + \cdots + \boldsymbol{X}_n\boldsymbol{X}_n^\top)/n$ might be degenerate, and therefore we will occasionally refer to the non-degenerate subspace $\boldsymbol{U}_r = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_r]$, where $\boldsymbol{U}$ is given by the Singular Value Decomposition (SVD): $\widehat{\boldsymbol{\Sigma}} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$ and $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_r$ are the eigenvectors corresponding to the eigenvalues $\hat{\lambda}_1, \ldots, \hat{\lambda}_r$, where $\hat{\lambda}_i = \lambda_i(\widehat{\boldsymbol{\Sigma}})$, arranged in decreasing order:

$$\lambda_1(\widehat{\boldsymbol{\Sigma}}) \geq \lambda_2(\widehat{\boldsymbol{\Sigma}}) \geq \cdots \geq \lambda_r(\widehat{\boldsymbol{\Sigma}}) > 0$$

and $r = \operatorname{rank}(\widehat{\boldsymbol{\Sigma}})$. We write $\widehat{\lambda}^+_{\min} = \lambda^+_{\min}(\widehat{\boldsymbol{\Sigma}}) = \lambda_r(\widehat{\boldsymbol{\Sigma}})$ for the minimal *non-zero* eigenvalue, and we denote $\widehat{\boldsymbol{M}} = \boldsymbol{U}_r\boldsymbol{U}_r^\top$. Note that $\widehat{\boldsymbol{M}}^2 = \widehat{\boldsymbol{M}}$. Now we state our main result in this setting.

Theorem 2. *Assume that $\boldsymbol{W}_0 \sim \mathcal{N}(\boldsymbol{0}, \nu^2_{\text{init}}\boldsymbol{I})$. Then, for any $\boldsymbol{w}^\star \in \arg\min_{\boldsymbol{w}\in\mathbb{R}^d} L(\boldsymbol{w})$ and any $x > 0$, with probability $1 - e^{-x}$ over random samples S we have*

$$\mathbb{E}[\mathcal{E}(\boldsymbol{w}^\star)] \leq \mathbb{E}\left[(1 - \alpha\widehat{\lambda}^+_{\min})^{2T}\right]\left(\|\boldsymbol{w}^\star\|^2 + \nu^2_{\text{init}}(2+d)\right) + \frac{1}{2}\,\mathbb{E}[\|\boldsymbol{w}^\star\|^2_{\boldsymbol{I}-\widehat{\boldsymbol{M}}}]\,.$$

The proof is in Appendix D. This is a consequence of Theorem 1, modulo showing that GD with the least squares objective is $(\Delta, \widehat{\boldsymbol{M}})$-admissible with $\Delta = (1 - \alpha\widehat{\lambda}^+_{\min})^T$, and upper-bounding $\mathbb{E}[\|\boldsymbol{w}^\star\|^2_{\boldsymbol{I}-\widehat{\boldsymbol{M}}}]$ by controlling the expected squared norm of the projection onto the orthogonal complement of the space spanned by $\boldsymbol{U}_r$. The later comes up in the analysis of PCA (see e.g. Shawe-Taylor et al. [2005, Theorem 1]), and as we show in Appendix F this term is expected to be small enough whenever the eigenvalues have exponential decay, in which case with high probability we have $\mathbb{E}[\|\boldsymbol{w}^\star\|^2_{\boldsymbol{I}-\widehat{\boldsymbol{M}}}] \lesssim \|\boldsymbol{w}^\star\|_2^2/\sqrt{n}$ as $n \to \infty$. Note that the middle term in the upper bound of our Theorem 1 vanishes in the noise-free case: $\mathbb{E}[\|\mathcal{A}_S(\boldsymbol{w}^\star) - \boldsymbol{w}^\star\|^2_{\widehat{\boldsymbol{M}}}] = 0$.

Looking at Theorem 2, we can see that the excess risk is bounded by the sum of two terms. Note that the second term is negligible in many cases (consider the limit of infinite data) and additionally it is a term that remains constant during training as it does not depend on training data. Therefore, we are particularly interested in the first term of the bound, which is data-dependent. This term depends on $\widehat{\lambda}^+_{\min}$ via a functional form that has a double descent behaviour if plotted against d for fixed n. Before going into that analysis, let us also consider the scenario with label noise.

2.2 Least-Squares with Random Design and Label Noise

Now, in addition to the random design we introduce label noise into our model:

$$Y = \boldsymbol{X}^\top\boldsymbol{w}^\star + \varepsilon\,,$$

where we have random noise ε such that $\mathbb{E}[\varepsilon] = 0$ and $\mathbb{E}[\varepsilon^2] = \sigma^2$, independent of the instances.

Theorem 3. *Assume that $\boldsymbol{W}_0 \sim \mathcal{N}(\boldsymbol{0}, \nu^2_{\text{init}}\boldsymbol{I})$. Then, for any $\boldsymbol{w}^\star \in \arg\min_{\boldsymbol{w}\in\mathbb{R}^d} L(\boldsymbol{w})$ and any $x > 0$, with probability $1 - e^{-x}$ over random samples S we have*

$$\mathbb{E}[\mathcal{E}(\boldsymbol{w}^\star)] \leq \mathbb{E}\left[(1 - \alpha\widehat{\lambda}^+_{\min})^{2T}\right]\left(\|\boldsymbol{w}^\star\|^2 + \nu^2_{\text{init}}(2+d)\right) + \frac{4\sigma^2}{n}\,\mathbb{E}\left[\left(\widehat{\lambda}^+_{\min}\right)^{-2}\right] + \frac{1}{2}\,\mathbb{E}[\|\boldsymbol{w}^\star\|^2_{\boldsymbol{I}-\widehat{\boldsymbol{M}}}]\,.$$

The proof is in Appendix D. Again, this follows from Theorem 1, by the same steps used in the proof of Theorem 2, except that the term $\mathbb{E}\left[\|\boldsymbol{w}^\star - \mathcal{A}_S(\boldsymbol{w}^\star)\|^2_{\widehat{\boldsymbol{M}}}\right]$ is now handled by conditioning on the sample and analyzing the expectation with respect to the random noise (Lemma 4 and its proof in Appendix D.2), leading to the new term $\frac{4\sigma^2}{n}\,\mathbb{E}\left[\left(\widehat{\lambda}^+_{\min}\right)^{-2}\right]$. The latter closely resembles the term one would get for ridge regression [Shalev-Shwartz and Ben-David, 2014, Cor. 13.7] due to algorithmic stability [Bousquet and Elisseeff, 2002], but here we have a dependence on the smallest non-zero eigenvalue instead of a regularization parameter.

2.3 Concentration of the Smallest Non-zero Eigenvalue

In this section we take a look at the behaviour of $\widehat{\lambda}^+_{\min}$ assuming that input instances $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ are i.i.d. *random* vectors, sampled from some underlying marginal density that meets some regularity

requirements (Definitions 2 and 3 below) so that we may use the results from random matrix theory [Vershynin, 2012]. Recall that the covariance matrix of the input features is $\widehat{\boldsymbol{\Sigma}} = (\boldsymbol{X}_1\boldsymbol{X}_1^\top + \cdots + \boldsymbol{X}_n\boldsymbol{X}_n^\top)/n$. We focus on the concentration of $\widehat{\lambda}^+_{\min} = \lambda^+_{\min}(\widehat{\boldsymbol{\Sigma}})$ around its population counterpart $\lambda^+_{\min} = \lambda^+_{\min}(\boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is the population covariance matrix: $\boldsymbol{\Sigma} = \mathbb{E}[\boldsymbol{X}_1\boldsymbol{X}_1^\top]$.

In particular, the Bai-Yin limit characterization of the extreme eigenvalues of sample covariance matrices [Bai and Yin, 1993] implies that $\widehat{\lambda}^+_{\min}$ has almost surely an asymptotic behavior $(1-\sqrt{d/n})^2$ as the dimensions grow to infinity, assuming that the matrix $\boldsymbol{X} := [\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n] \in \mathbb{R}^{d\times n}$ has independent entries. We are interested in the non-asymptotic version of this result. However, unlike Bai and Yin [1993], we do not assume independence of all entries, but rather independence of observation vectors (columns of $\boldsymbol{X}$). This will be done by introducing a distributional assumption: we assume that observations are *sub-Gaussian* and *isotropic* random vectors.

Definition 2 (Sub-Gaussian random vectors)**.** *A random vector $\boldsymbol{X} \in \mathbb{R}^d$ is sub-Gaussian if the random variables $\boldsymbol{X}^\top\boldsymbol{y}$ are sub-Gaussian for all $\boldsymbol{y} \in \mathbb{R}^d$. The sub-Gaussian norm of a random vector $\boldsymbol{X} \in \mathbb{R}^d$ is defined as*

$$\|\boldsymbol{X}\|_{\psi_2} = \sup_{\|\boldsymbol{y}\|=1} \sup_{p\geq 1} \left\{ \frac{1}{\sqrt{p}} \mathbb{E}[|\boldsymbol{X}^\top\boldsymbol{y}|^p]^{\frac{1}{p}} \right\} .$$

Definition 3 (Isotropic random vectors)**.** *A random vector $\boldsymbol{X} \in \mathbb{R}^d$ is called isotropic if its covariance is the identity: $\mathbb{E}\left[\boldsymbol{X}\boldsymbol{X}^\top\right] = \boldsymbol{I}$. Equivalently, $\boldsymbol{X}$ is isotropic if $\mathbb{E}[(\boldsymbol{X}^\top\boldsymbol{x})^2] = \|\boldsymbol{x}\|^2$ for all $\boldsymbol{x} \in \mathbb{R}^d$.*

Let $\boldsymbol{\Sigma}^\dagger$ be the Moore-Penrose pseudoinverse of $\boldsymbol{\Sigma}$. In Appendix E we prove the following.[1]

Lemma 1 (Smallest non-zero eigenvalue of sample covariance matrix)**.** *Let $\boldsymbol{X} = [\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n] \in \mathbb{R}^{d\times n}$ be a matrix with i.i.d. columns, such that $\max_i \|\boldsymbol{X}_i\|_{\psi_2} \leq K$, and let $\widehat{\boldsymbol{\Sigma}} = \boldsymbol{X}\boldsymbol{X}^\top/n$, and $\boldsymbol{\Sigma} = \mathbb{E}[\boldsymbol{X}_1\boldsymbol{X}_1^\top]$. Then, for every $x \geq 0$, with probability at least $1 - 2e^{-x}$, we have*

$$\lambda^+_{\min}(\widehat{\boldsymbol{\Sigma}}) \geq \lambda^+_{\min}(\boldsymbol{\Sigma}) \left(1 - K^2\left(c\sqrt{\frac{d}{n}} + \sqrt{\frac{x}{n}}\right)\right)^2_+ \qquad \text{for } n \geq d\,,$$

and furthermore, assuming that $\|\boldsymbol{X}_i\|_{\boldsymbol{\Sigma}^\dagger} = \sqrt{d}$ a.s. for all $i \in [n]$, we have

$$\lambda^+_{\min}(\widehat{\boldsymbol{\Sigma}}) \geq \lambda^+_{\min}(\boldsymbol{\Sigma}) \left(\sqrt{\frac{d}{n}} - K^2\left(c + 6\sqrt{\frac{x}{n}}\right)\right)^2_+ \qquad \text{for } n < d\,,$$

where we have an absolute constant $c = 2^{3.5}\sqrt{\ln(9)}$.

Lemma 1 is a non-asymptotic result that allows us to understand the behaviour of $\widehat{\lambda}^+_{\min}$, and hence the behaviour of the excess risk that depends on this quantity, for fixed dimensions. We will exploit this fact in the following section in which we discuss the implications of our findings.

3 Excess risk as a function of over-parammetrization

First we note that, in the noise-free case, the middle term in the upper bound of Theorem 1 vanishes: $\mathbb{E}[\|\mathcal{A}_S(\boldsymbol{w}^\star) - \boldsymbol{w}^\star\|^2_{\widehat{\boldsymbol{M}}}] = 0$. Thus, as in Theorem 2, the upper bound consists only of the term involving the smallest positive eigenvalue $\widehat{\lambda}^+_{\min}$ and the term involving $\mathbb{E}[\|\boldsymbol{w}^\star\|^2_{\boldsymbol{I}-\widehat{\boldsymbol{M}}}]$. The behaviour of the former was clarified in Section 2.3, and the latter is controlled as explained in Appendix F. Thus, in the *overparametrized* regime ($d > n$) we have:

$$\mathbb{E}[\mathcal{E}(\boldsymbol{w}^\star)] \lesssim \left(1 - \frac{\alpha}{n}(\sqrt{d} - \sqrt{n} - 1)^2_+\right)^{2T} \|\boldsymbol{w}^\star\|^2 + \mathbb{E}\left[\|\boldsymbol{w}^\star\|^2_{\boldsymbol{I}-\widehat{\boldsymbol{M}}}\right] .$$

A similar bound holds in the *underparametrized* case ($d < n$) but replacing the term $(\sqrt{d} - \sqrt{n} - 1)^2_+$ with $(\sqrt{n} - \sqrt{d} - 1)^2_+$. Note that the term multiplying the learning rate is $(\sqrt{d/n} - 1 - 1/\sqrt{n})^2$,

[1] $(x)_+ = \max\{0, x\}$

in accordance with the Bai-Yin limit which says that asymptotically $\widehat{\lambda}^{+}_{\min} \sim (\sqrt{d/n} - 1)^2$. It is interesting to see how $\left(1 - \alpha(\sqrt{d/n} - 1)^2_{+}\right)^{2T}$ varies with model size d for a given fixed dataset size n and fixed number of gradient updates T. Setting $y = d/n$ and considering the cases $y \to 0$ (underparametrized regime), $y \sim 1$ (the peak), and $y > 1$ (overparametrized regime) it becomes evident that this term has a double descent behaviour. Thus, the double descent is captured in the part of the excess risk bound that corresponds to learning dynamics on the space spanned by $\widehat{\boldsymbol{M}}$.

Similarly, we can now consider the scenario with label noise: we can similarly bound the excess risk, following the same logic as for noise-free case; however we have an additional dependence on σ^2 via the term $\frac{4\sigma^2}{n}\,\mathbb{E}\left[\left(\widehat{\lambda}^{+}_{\min}\right)^{-2}\right]$. While this does not interfere with the DD shape as we change model size, it does imply that the peak is dependent on the amount of noise. In particular, the more noise we have in the learning problem the larger we expect the peak at the interpolation boundary to be.

While the presence of the double descent has been studied by several works, our derivation provides two potentially new interesting insights. The first one is that there is a dependency between the noise in the learning problem and the shape of the curve, the larger the noise is, the larger the peak in DD curve. This agrees with the typical intuition in the underparmetrized regime that the model fits the noise when it has enough capacity, leading towards a spike in test error. However, due to the dependence on $\widehat{\lambda}^{+}_{\min}$, it is subdued as the model size grows. Secondly, and maybe considerably more interesting, there seems to be a connection between the double descent curve of the excess risk and the optimization process. In particular, our derivation is specific to gradient descent. In this case the excess risk seems to depend on the conditioning of the features in the least squares problem on the subspace spanned by the data through $\widehat{\lambda}^{+}_{\min}$, which also affects convergence of the optimization process. For the least squares problem this can easily be seen, as the sample covariance of the features corresponds to the Gauss-Newton approximation of the Hessian [e.g. Nocedal and Wright, 2006], hence it impacts the convergence. In a more precise way, conditioning of any matrix is measured by the ratio $s_{\max}/s_{\min}$ (the 'condition number') which is determined solely by the smallest singular value $s_{\min}$ in cases when $s_{\max}$ is of constant order, such as the case that we studied here: Note that by our boundedness assumption, $s_{\max}$ is constant, but in general one needs to consider both $s_{\max}$ and $s_{\min}$ in order to characterize the condition numbers, which interestingly have been observed to display a double descent as well Poggio et al. [2019].

More generally, normalization, standardization, whitening and various other preprocessing of the input data have been a default step in many computer vision systems [e.g. LeCun et al., 1998, Krizhevsky, 2009] where it has been shown empirically that they greatly affect learning. Such preprocessing techniques are usually aimed to improve conditioning of the data. Furthermore, various normalization layers like batch-norm [Ioffe and Szegedy, 2015] or layer-norm [Ba et al., 2016] are typical components of recent architectures, ensuring that features of intermediary layers are well conditioned. Furthermore, it has been suggested that model size improves conditioning of the learning problem [Li et al., 2018], which is in line with our expectation given the behaviour of $\widehat{\lambda}^{+}_{\min}$. Taking inspiration from the optimization literature, it is natural for us to ask whether for neural networks, we can also connect the conditioning or $\widehat{\lambda}^{+}_{\min}$ of intermediary features and double descent. This particular might be significant if we think of the last layer of the architecture as a least squares problem (assuming we are working with mean square error), and all previous layers as some random projection, ignoring that learning is affecting this projection as well.

This relationship between generalization and double descent on one hand, and the conditioning of the features and optimization process raises some additional interesting questions, particularly since, compared to the typical least squares setting, the conditioning of the problem for deep architectures does not solely depend on size. In the next section we empirically look at some of these questions.

4 Empirical exploration in neural networks

The first natural question to ask is whether the observed behaviour for the least squares problem is reflected when working with neural networks. To explore this hypothesis, and to allow tractability of computing various quantities of interest (like $\widehat{\lambda}^{+}_{\min}$), we focus on one hidden layer MLPs on the MNIST and FashionMNIST datasets. We follow the protocol used by Belkin et al. [2019], relying on a squared error loss. In order to increase the model size we simply increase the dimensionality

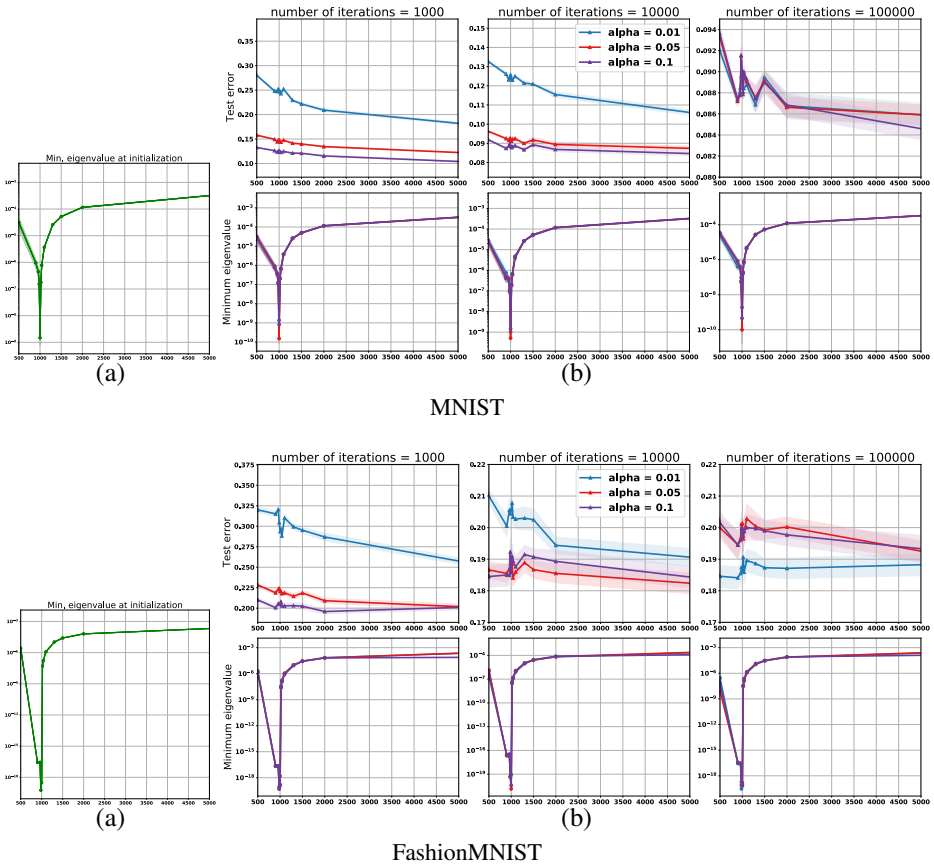

Figure 2: Training one hidden layer networks of increasing width on MNIST (top) and FashionMNIST (bottom): (a) Minimum positive eigenvalue of the intermediary features at initialization - (b) Test error and corresponding minimum eigenvalue of the intermediary features at different iterations

of the latent space, and rely on gradient descent with a fixed learning rate and a training set to 1000 randomly chosen examples for both datasets. More details can be found in Appendix H.

Figure 2 provides the main findings on this experiment. Similar to the Figure 1, we depict 3 columns showing snapshots at different number of gradient updates: 1000, 10000 and 100000. The first row shows test error (number of miss-classified examples out of the test examples) computed on the full test set of 10000 data points which as expected shows the double descent curve with a peak around 1000 hidden units. Note that the peak is relatively small, however the behaviour seems consistent under 5 random seeds for the MNIST experiment.[2] The second row and potentially the more interesting one looks at the $\widehat{\lambda}^{+}_{\min}$ computed on the covariance of the activations of the hidden layer, which as predicted by our theoretical derivation shows a dip around the interpolation threshold, giving the expected U-shape. Even more surprisingly this shape seems to be robust throughout learning, and the fact that the input weights and biases are being trained seems not to alter it, thus suggesting that our derivation might provide insights in the behaviour of deep models.

Following this, if we think of the output layer as solving a least squares problem, while the rest of the network provides a projection of the data, we can consider what can affect the conditioning of the last latent space of the network. We put forward the hypothesis that $\widehat{\lambda}^{+}_{\min}$ is not simply affected by the number of parameters, but actually the distribution of these parameters in the architecture matters.

To test this hypothesis, we conduct an experiment where we compare the behavior of a network with a single hidden layer and a network with three hidden layers. For both networks, we increase the size of the hidden layers. For the deeper network, we consider either increasing the size of all the hidden

[2]The error bars for the test error in all the other experiments are estimated by splitting the test set into 10 subsets.

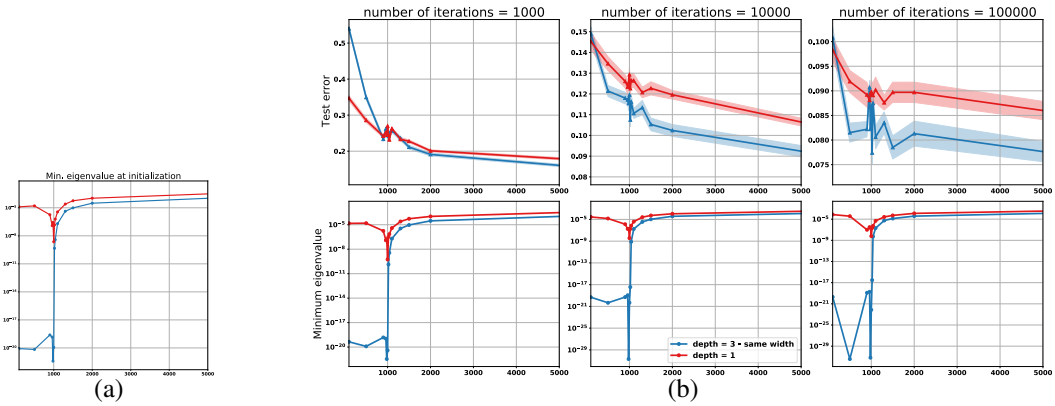

Figure 3: Training networks of increasing width with 1 and 3 hidden layers on MNIST: (a) Minimum positive eigenvalue of the intermediary features at initialization - (b) Test error and corresponding minimum eigenvalue of the intermediary features at different iterations

layers or grow only the last hidden layer while keeping the others to a fixed small size, creating a strong bottleneck in the network. Figure 3 shows the results obtained with the former, while the effect of the bottleneck can be seen in Appendix G. We first observe that for the three tested networks, the drop in the minimum eigenvalues happens when the size of the last hidden layer reaches the number of training samples, as predicted by the theory. The magnitude of this drop and behavior across the different tested sizes depends however on the previous layers. In particular, we observe that the bottleneck yields features that are more ill-conditioned than the network with wide hidden layers, where the width of the last layer on its own can not compensate for the existence of the bottleneck. Moreover, from Figure 3, we can clearly see that the features obtained by the deeper network have a bigger drop in the minimum eigenvalue, which results, as expected in a higher increase in the test error around the interpolation threshold.

It is well known that depth can harm optimization making the problem ill-conditioned, hence the reliance on skip-connections and batch normalization De and Smith [2020] to train very deep architecture. Our construction provides a way of reasoning about double descent that allows us to factor in the ill-conditioning of the learning problem. Rather than focusing simply on the model size, it suggests that for neural networks the quantity of interest might also be $\widehat{\lambda}^{+}_{\min}$ for intermediary features, which is affected by size of the model but also by the distribution of the weights and architectural choices. For now we present more empirical explorations and ablations in Appendix H, and put forward this perspective as a conjecture for further exploration.

5 Conclusion and Future Work

In this work we analyse the double descent phenomenon in the context of the least squares problem. We make the observation that the excess risk of gradient descent is controlled by the smallest *positive* eigenvalue, $\widehat{\lambda}^{+}_{\min}$, of the feature covariance matrix. Furthermore, this quantity follows the Bai-Yin law with high probability under mild distributional assumptions on features, that is, it manifests a U-shaped behaviour as the number of features increases, which we argue induces a double descent shape of the excess risk. Through this we provide a connection between the widely known phenomena and optimization process and conditioning of the problem. We believe this insight provides a different perspective compared to existing results focusing on the Moore-Penrose pseudo-inverse solution. In particular our work conjectures that the connection between the known double descent shape and model size is through $\widehat{\lambda}^{+}_{\min}$ of the features at intermediary layers. For the least squares problem $\widehat{\lambda}^{+}_{\min}$ correlates strongly with model size (and hence feature size). However this might not necessarily be always true for neural networks. For example we show empirically that while both depth and width increase the model size, they might affect $\widehat{\lambda}^{+}_{\min}$ differently. We believe that our work could enable much needed effort, either empirical or theoretical, to disentangle further the role of various factors, like depth and width or other architectural choices like skip connections on double descent.

References

Andrew Brock, Soham De, Samuel L Smith, and Karen Simonyan. High-performance large-scale image recognition without normalization. arXiv:2102.06171, 2021.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Andrew W Senior, Richard Evans, John Jumper, James Kirkpatrick, Laurent Sifre, Tim Green, Chongli Qin, Augustin Žídek, Alexander WR Nelson, Alex Bridgland, et al. Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792):706–710, 2020.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2 edition, 2009.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019. Previously arXiv:1812.11118.

Mikhail Belkin, Daniel Hsu, and Ji Xu. Two models of double descent for weak features. *SIAM Journal on Mathematics of Data Science*, 2(4):1167–1180, 2020. Accessed from arXiv:1903.07571.

László Györfi, Michael Kohler, Adam Krzyzak, and Harro Walk. *A distribution-free theory of nonparametric regression*, volume 1. Springer, 2002.

Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and double descent curve. arXiv:1908.05355, 2019.

Michal Derezinski, Feynman T Liang, and Michael W Mahoney. Exact expressions for double descent and implicit regularization via surrogate random design. In *Advances in Neural Information Processing Systems [NeurIPS 2020]*, 2020.

John Shawe-Taylor, Christopher KI Williams, Nello Cristianini, and Jaz Kandola. On the eigenspectrum of the gram matrix and the generalization error of kernel-pca. *IEEE Transactions on Information Theory*, 51(7):2510–2522, 2005.

Olivier Bousquet and André Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2:499–526, 2002.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. In *Compressed Sensing, Theory and Applications*, pages 210–268. Cambridge University Press, 2012. Accessed from arXiv:1011.3027.

Zhi-Dong Bai and Yong-Qua Yin. Limit of the smallest eigenvalue of a large dimensional sample covariance matrix. *The Annals of Probability*, 21(3):1275–1294, 1993.

Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, second edition, 2006.

Tomaso Poggio, Gil Kur, and Andrzej Banburski. Double descent in the condition number. Technical Report CBMM Memo No. 102, MIT, 2019. Accessed from arXiv:1912.06190.

Yann LeCun, Léon Bottou, Genevieve B. Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural Networks: Tricks of the Trade (2nd ed.)*, Lecture Notes in Computer Science, pages 9–48. Springer, 1998.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 448–456. JMLR.org, 2015.

Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *CoRR*, abs/1607.06450, 2016.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Soham De and Sam Smith. Batch normalization biases residual blocks towards the identity function in deep networks. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 19964–19975. Curran Associates, Inc., 2020.

Akshay Rangamani, Lorenzo Rosasco, and Tomaso Poggio. For interpolating kernel machines, minimizing the norm of the erm solution minimizes stability, 2020.

Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. arXiv:1903.08560, 2019.

Ben Adlam and Jeffrey Pennington. Understanding double descent requires a fine-grained bias-variance decomposition. In *Advances in Neural Information Processing Systems [NeurIPS 2020]*, 2020.

Peter L Bartlett, Andrea Montanari, and Alexander Rakhlin. Deep learning: a statistical viewpoint. arXiv:2103.09177, 2021.

Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. In *International Conference on Learning Representations*, 2019.

Tengyuan Liang, Alexander Rakhlin, et al. Just interpolate: Kernel "ridgeless" regression can generalize. *Annals of Statistics*, 48(3):1329–1347, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] These are summarised in Section 5
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] We believe that presented research should be categorized as basic research and we are not targeting any specific application area. Theorems may inspire new algorithms and theoretical investigation. The algorithms presented here can be used for many different applications and a particular use may have both positive or negative impacts. We are not aware of any immediate short term negative implications of this research and we believe that a broader impact statement is not required for this paper.

(d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] Assumptions are stated in all statements of theorems.
   (b) Did you include complete proofs of all theoretical results? [Yes] The main paper presents proof sketch while all the proofs with details are deferred to the supplementary material.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Instructions for experiments are presented in Section 4
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] Synthetic experiments presented in the introduction can be run on any contemporary laptop, resources are also discussed in Section 4.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]