Small random initialization is akin to spectral learning: Optimization and generalization guarantees for overparameterized low-rank matrix reconstruction

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recently there has been significant theoretical progress on understanding the convergence and generalization of gradient-based methods on nonconvex losses with overparameterized models. Nevertheless, many aspects of optimization and generalization and in particular the critical role of small random initialization are not fully understood. In this paper, we take a step towards demystifying this role by proving that small random initialization followed by a few iterations of gradient descent behaves akin to popular spectral methods. We also show that this *implicit spectral bias* from small random initialization, which is provably more prominent for overparameterized models, also puts the gradient descent iterations on a particular trajectory towards solutions that are not only globally optimal but also generalize well. Concretely, we focus on the problem of reconstructing a low-rank matrix from a few measurements via a natural nonconvex formulation. In this setting, we show that the trajectory of the gradient descent iterations from small random initialization can be approximately decomposed into three phases: (I) a *spectral or alignment phase* where we show that that the iterates have an implicit spectral bias akin to spectral initialization allowing us to show that at the end of this phase the column space of the iterates and the underlying low-rank matrix are sufficiently aligned, (II) a *saddle avoidance/refinement phase* where we show that the trajectory of the gradient iterates moves away from certain degenerate saddle points, and (III) a *local refinement phase* where we show that after avoiding the saddles the iterates converge quickly to the underlying low-rank matrix. Underlying our analysis are insights for the analysis of overparameterized nonconvex optimization schemes that may have implications for computational problems beyond low-rank reconstruction.

1 Introduction

Many contemporary problems in machine learning and signal estimation spanning deep learning to low-rank matrix reconstruction involve fitting nonlinear models to training data. Despite tremendous empirical progress, theoretical understanding of these problems poses two fundamental challenges. First, from an *optimization* perspective, fitting these models often requires solving highly nonconvex optimization problems and except for a few special cases, it is not known how to provably find globally or approximately optimal solutions. Yet simple heuristics such as running (stochastic) gradient descent from (typically) small random initialization is surprisingly effective at finding globally optimal solutions. A second *generalization* challenge is that many modern learning models including

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

neural network architectures are trained in an overparameterized regime where the parameters of the model exceed the size of the training dataset. It is well understood that in this overparameterized regime, these large models are highly expressive and have the capacity to (over)fit arbitrary training datasets including pure noise. Mysteriously however overparameterized models trained via simple algorithms such as (stochastic) gradient descent when initialized at random continue to predict well or generalize on yet unseen test data. Indeed small random initialization followed by (stochastic) gradient descent iterative updates is arguably the most widely used learning algorithm in modern machine learning and signal estimation.

There has been a large number of exciting results aimed at demystifying both the optimization and generalization aspects over the past few years. We will elaborate on these results in detail in the supplementary, however, we would like to briefly mention the common techniques and their existing limitations. On the optimization front a large body of work has emerged on providing guarantees for nonconvex optimization which can roughly be put into two categories: (I) smart initialization+local convergence and (II) landscape analysis+saddle escaping algorithms. Approaches in (I) focus on showing local convergence of local search techniques from carefully designed spectral initializations [1, 2, 3, 4, 5, 6, 7, 8]. Approaches in (II) focus on showing that in some cases the optimization landscape is benign in the sense that all local minima are global (no spurious local minima) and the saddle points have a direction of strict negative curvature (strict saddle) [9]. Then specialized truncation or saddle escaping algorithms such as trust region, cubic regularization [10, 11], or noisy (stochastic) gradient-based methods [12, 13, 14, 15] are deployed to provably find a global optimum. Both approaches fail to fully explain the typical behavior of local search techniques in practice. Indeed, for many nonconvex problems local search techniques or simple variants, when initialized at random, quickly converge to globally optimal solutions without getting stuck in local optima/saddles without the need for sophisticated initialization or saddle escaping heuristics. We note that while for differentiable losses eventual convergence to local minimizers is known from a random initialization [16] on problems of the form (II), these results cannot rule out exponentially slow cases in the worst-case [17]. Indeed, it has been argued that a more granular analysis of the trajectory of gradient descent beyond the landscape may be necessary [18].

Similarly, there has been a lot of exciting progress on the generalization front, especially for neural networks. Specific to generalization capabilities of gradient-based approaches these results broadly fall into two categories: (1) the first category is based on a linearization principle which characterizes the performance of nonlinear models such as neural networks by comparing it to a linearized kernel problem around the initialization (a.k.a. Neural Tangent Kernels) [19, 20, 21, 22, 23]. (2) the second category is based on a continuous limit analysis in the limit of width going to infinity and learning rate going to zero (mean-field analysis) [24, 25, 26, 27, 28]. However, these existing analyses contain many idealized and non-realistic assumptions (e.g. requiring large widths or large standard deviation at random initialization) and therefore cannot fully explain the success of overparameterized models or serve as a guiding principle for practitioners [29].

Despite the aforementioned exciting recent theoretical progress many aspects of optimization and generalization and in particular the role of random initialization remains mysterious. This leads us to the main challenge of this paper

> *Why is small random initialization combined with gradient descent updates so effective at finding globally optimal models that generalize well despite the nonconvex nature of the optimization landscape or model overparameterization?*

In this paper we wish to take a step towards addressing the above challenge by demystifying the critical role of small random initialization in gradient-based approaches. Specifically we show that

> *Small random initialization followed by a few iterations of gradient descent behaves akin to spectral initialization.*

By that, we mean more precisely, that if the initialization is chosen small enough, then in the initial stage of the training, gradient descent *implicitly* behaves like spectral initialization techniques such as those commonly used in techniques based on the method of moments. This *implicit spectral bias* of gradient descent from random initialization puts the gradient descent iterations on a particular trajectory towards solutions that are not only globally optimal but also generalize well for overparameterized models. We also show that with small random initialization this implicit spectral bias

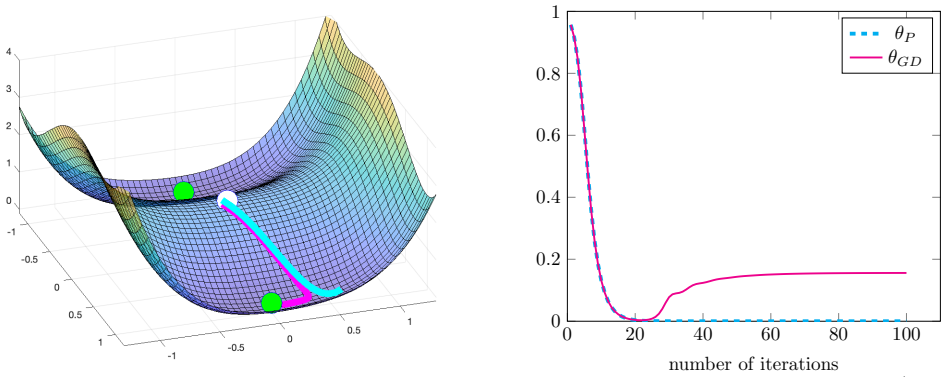


Figure 1: **Gradient descent from small random initialization is akin to spectral initialization.** The left figure depicts the empirical landscape of a low-rank matrix reconstruction problem with the two green circles depicting the two global minima and the white circle the saddle point at the origin. In this figure, we also depict the trajectory of the gradient descent iterations (magenta) together with the power method based on a popular spectral initialization technique (blue). Both gradient descent and power method use the same small initialization near the origin. We see that in the early stage, the two trajectories are almost the same. The figure on the right depicts the angle between the gradient descent (magenta)/power method (blue) iterates and a popular spectral initialization technique, denoted by θ_{GD} and θ_P respectively. This figure clearly demonstrates that for the first iterations these angles are practically the same further confirming that the initial trajectory of gradient descent and power methods are similar. See Section 5 for further detail on the experimental setup.

phenomenon is more prominent for more overparameterized models in the sense that it materializes after fewer iterations. This intriguing phenomenon is depicted in Figure 1 in the context of a low-rank reconstruction problem. This figure clearly demonstrates that the first few iterations of gradient descent starting from a small random initialization are virtually identical to that of running power iterations (a popular algorithm to find the spectral initialization).

Concretely we focus on the problem of low-rank matrix recovery, where our goal is to recover a low-rank matrix of the form XX^T from a few linear measurements. We consider a natural, non-convex approach based on matrix factorization, where we minimize the loss function via gradient descent. In this paper, we show that, regardless of the amount of overparameterization used, for small random initialization vanilla gradient descent will always converge towards the low-rank solution. This holds as long the measurement operator obeys a popular restricted isometry property [30].[1] Our analysis consists of three phases. The first phase is the aforementioned *spectral* or *alignment* phase where we show gradient descent from small random initialization behaves akin to spectral initialization. Next, we show that after this first *spectral or alignment phase*, gradient descent enters a second phase, which we refer to as *saddle avoidance phase*. In this phase, we show that the trajectory of the gradient iterates moves away from degenerate saddle points, while the iterates maintain almost the same effective rank as XX^T. In the third phase, the *local refinement phase*, we show that the iterates approximately converge towards the underlying low-rank matrix XX^T with a geometric rate up to a certain error floor which depends on the initialization scale. In particular, by decreasing the scale of initialization this error threshold can be made arbitrarily small. While in this paper our main focus is on low-rank matrix reconstruction, we believe that our analysis holds more generally for a variety of contemporary machine learning and signal estimation tasks including neural networks.

2 Low-rank matrix recovery via non-convex optimization

As mentioned earlier in this paper we focus on reconstructing a (possibly overparameterized) Positive Semidefinite (PSD) low rank matrix from a few measurements. In this problem, given m observations

[1] To the best of our knowledge, even for the non-overparameterized scenario, this is the first result that shows that vanilla gradient descent converges to the ground truth from a random initialization, when the measurement operator only satisfies the restricted isometry property.

of the form

$$y_i = \langle A_i, XX^T \rangle = \mathrm{Tr}\left(A_i XX^T\right) \qquad i = 1, \ldots, m, \tag{1}$$

we wish to reconstruct the unknown matrix XX^T. Here, $X \in \mathbb{R}^{n \times r_\star}$ with $1 \le r_\star \le n$ is a factor of the unknown matrix and $\{A_i\}_{i=1}^m$ are known symmetric measurement matrices. A common approach to solving this problem is via minimizing the loss function

$$\min_{\bar{U} \in \mathbb{R}^{n \times r}} f(\bar{U}) := \min_{\bar{U} \in \mathbb{R}^{n \times r}} \frac{1}{4m} \sum_{i=1}^{m} \left(y_i - \langle A_i, \bar{U}\bar{U}^T \rangle\right)^2,$$

with $r \ge r_\star$. More compactly one can rewrite the optimization problem above in the form

$$\min_{\bar{U} \in \mathbb{R}^{n \times r}} f(\bar{U}) := \min_{\bar{U} \in \mathbb{R}^{n \times r}} \frac{1}{4} \left\| \mathcal{A}\left(\bar{U}\bar{U}^T - XX^T\right) \right\|_{\ell_2}^2, \tag{2}$$

where $\mathcal{A} : \mathbb{R}^{n \times n} \longrightarrow \mathbb{R}^m$ is the measurement operator defined by $\left[\mathcal{A}(Z)\right]_i := \frac{1}{\sqrt{m}} \langle A_i, Z \rangle$.

In order to solve the minimization problem (2) we run gradient descent iterations starting from (often small) random initialization. More specifically,

$$U_{t+1} = U_t - \mu \nabla f(U_t) = U_t + \mu \left[\left(\mathcal{A}^* \mathcal{A}\right) \left(XX^T - U_t U_t^T\right) \right] U_t.$$

where $U_0 = \alpha U$ is the initialization matrix. Here, $U \in \mathbb{R}^{n \times r}$ is a typically random matrix which represents the form of the initialization and $\alpha > 0$ is a scaling parameter.

There are two challenges associated with analyzing such randomly initialized gradient descent updates. The first is an *optimization* challenge. Since f is *non-convex* it is a priori not clear whether gradient descent converges to a global optimum or whether it gets stuck in a local minima and/or saddle. The second challenge is that of *generalization*. This is particularly pronounced in the overparameterized scenario where the number of parameters are larger than the number of data points i.e. $rn \ge m$. In this case, there are infinitely many $\bar{U}$ such that $f(\bar{U}) = 0$, but $\|\bar{U}\bar{U}^T - XX^T\|_F$ is arbitrarily large (see, e.g., [31, Proposition 1]). That is, even if gradient descent converges to a global optimum, i.e. $f\left(\bar{U}\right) = 0$, it is a priori not clear whether it has found the low-rank solution XX^T.

3 Main results

In this section, we present our main results. Stating these results requires a couple of simple definitions. The first definition concerns the measurement operator $\mathcal{A}$.

Definition 3.1 (Restricted Isometry Property (RIP))**.** *The measurement operator $\mathcal{A} : \mathbb{R}^{n \times n} \longrightarrow \mathbb{R}^m$ satisfies RIP of rank r with constant $\delta > 0$, if it holds for all matrices Z of rank at most r*

$$(1 - \delta) \|Z\|_F^2 \le \|\mathcal{A}(Z)\|_{\ell_2}^2 \le (1 + \delta) \|Z\|_F^2. \tag{3}$$

We note that for a Gaussian measurement operator $\mathcal{A}$ [2], RIP of rank r and constant $\delta > 0$ holds with high probability, if the number of observations satisfies $m \gtrsim nr/\delta^2$ [30, 32].

The second definition concerns the condition number of the factor X.

Definition 3.2 (condition number)**.** *We denote the condition number of $X \in \mathbb{R}^{n \times r_\star}$ by $\kappa := \frac{\|X\|}{\sigma_{r_\star}(X)}$, where $\sigma_{r_\star}(X)$ denotes $r_\star$-th largest singular value of X.*

With these definitions in place we are now ready to state our main results. Due to space limitations in the main paper we focus on the case of $r \ge 2r_\star$. With refer the reader to the supplementary for results covering all $r \ge r_\star$ including two special cases: (1) the fully overparameterized case, i.e., $r = n$ along with comparisons with existing work, in this case [36], and (2) the scenario that U has the same number of parameters as X, i.e., $r = r_\star$.

Theorem 3.3. *Let $X \in \mathbb{R}^{n \times r_\star}$ and assume we have m measurements of the low rank matrix XX^T of the form $y = \mathcal{A}\left(XX^T\right)$ with $\mathcal{A}$ the measurement operator. We assume $\mathcal{A}$ satisfies the restricted isometry property for all matrices of rank at most $2r_\star + 1$ with constant $\delta \le c\kappa^{-4} {r_\star}^{-1/2}$. To reconstruct*

[2] By that, we mean that all the entries of the (symmetric) measurement matrices $\{A_i\}_{i=1}^m$ are drawn i.i.d. with distribution $\mathcal{N}(0,1)$ on the off-diagonal and distribution $\mathcal{N}\left(0, 1/\sqrt{2}\right)$ on the diagonal.

XX^T from the measurements we fit a model of the form $\bar{U} \mapsto \mathcal{A}\left(\bar{U}\bar{U}^T\right)$ with $\bar{U} \in \mathbb{R}^{n\times r}$ and $r > r_*$ via running gradient descent iterations of the form $U_{t+1} = U_t - \mu\nabla f\left(U_t\right)$ on the objective (2) *with a step size obeying $\mu \leq c\kappa^{-4}\|X\|^{-2}$. Here, the initialization is given by $U_0 = \alpha U$, where $U \in \mathbb{R}^{n\times r}$ has i.i.d. entries distributed as $\mathcal{N}\left(0, 1/\sqrt{r}\right)$. Furthermore, we assume $r \geq 2r_\star$ and that the scale of initialization fulfills*

$$\alpha \lesssim \min\left\{\frac{\left(\min\{r;n\}\right)^{1/2}}{\kappa^{1/2}n^{3/4}}\left(2\kappa^2\sqrt{\frac{n}{\min\{r;n\}}}\right)^{-6\kappa^2}; \frac{\min\{r;n\}}{\kappa^7 n^2}\right\}\|X\|. \tag{4}$$

Then, after

$$\hat{t} \lesssim \frac{1}{\mu\sigma_{\min}\left(X\right)^2}\ln\left(\frac{C_1 n\kappa}{\min\{r;n\}}\cdot\max\left\{1; \frac{\kappa r_\star}{\min\{r;n\} - r_\star}\right\}\cdot\frac{\|X\|}{\alpha}\right)$$

iterations we have that

$$\frac{\|U_{\hat{t}}U_{\hat{t}}^T - XX^T\|_F}{\|X\|^2} \lesssim \frac{n^2\kappa^{81/16}r_\star^{1/8}}{\left(\min\{r;n\}\right)^{5/8}}\cdot\frac{\alpha^{21/16}}{\|X\|^{21/16}}, \tag{5}$$

holds with probability at least $1 - Ce^{-\tilde{c}r}$. Here, $C_1, C_2, \tilde{C}, \tilde{c} > 0$ are fixed numerical constants.

Note that the test error $\|U_{\hat{t}}U_{\hat{t}}^T - XX^T\|_F^2$ can be made arbitrarily small by choosing the scale of initialization α small enough. In particular, the dependence of the test error on α is polynomial and the dependence of the number of iterations on α is logarithmic, which means that reducing the test error by scaling down α introduces only modest additional computational cost. Hence, as long as the rank at most $2r_\star + 1$ RIP with constant $\delta \leq c\kappa^{-4}{r_\star}^{-1/2}$ holds, gradient descent converges to a point in the proximity of the low-rank solution, whenever the initialization is chosen small enough regardless of the choice of r. This holds even when the model is overparameterized i.e. $rn \gg m$ and the optimization problem has many global optima many of which do not obey $UU^T \approx XX^T$. This result thus further demonstrates that when initialized with a small random initialization gradient descent has an implicit bias towards solutions of low-rank or small nuclear norm. This is in sharp contrast to Neural Tangent Kernel (NTK)-based theory for low-rank matrix recovery (see [20, Section 4.2]) which will not approximately recover the ground truth matrix XX^T due to the larger scale of initialization required when using that technique.

As discussed in Section 2, the restricted isometry property holds with high probability for a sample complexity $m \gtrsim nr_\star^2\kappa^8$ for Gaussian measurement matrices. Up to constants, this sample complexity is optimal in n, while it is sub-optimal in $r_\star$ and κ compared to approaches based on nuclear-norm minimization (see, e.g., [30]). While there is numerical evidence that the true scaling of m in $r_\star$ should also be linear in the non-convex case [33], we note that the optimal dependence of the sample complexity on $r_\star$ is a major open problem in the field, as the sample complexities in all theoretical results for non-convex approaches in the literature scale at least quadratically in $r_\star$.

Interpretation: Recall from Section 1 that our convergence analysis can be divided into three phases: the spectral phase, the saddle avoidance phase, and the local refinement phase. As it will become clear from the proofs in the supplementary when $r \geq 2r_\star$ the bound on the number of iterations can be decomposed as follows

$$\hat{t} \lesssim \frac{1}{\mu\sigma_{\min}\left(X\right)^2}\left[\underbrace{\ln\left(2\kappa^2\sqrt{\frac{n}{\min\{r;n\}}}\right)}_{\text{Phase I: spectral/alignment phase}} + \underbrace{\ln\left(\frac{\sigma_{\min}\left(X\right)}{\alpha}\right)}_{\text{Phase II: saddle avoidance phase}} + \underbrace{\ln\left(\max\left\{1; \frac{\kappa r_\star}{\min\{r;n\} - r_\star}\right\}\frac{\|X\|}{\alpha}\right)}_{\text{Phase III: local refinement phase}}\right]. \tag{6}$$

First, we note that the duration of all three phases scales inversely with $\sigma_{\min}\left(X\right)^2$. This is due to the fact that in all three phases the dynamics associated the smallest singular value of X is the slowest one and hence needs the most time to complete.

In the spectral phase, the eigenvectors corresponding to the leading $r_\star$ eigenvalues of $U_tU_t^T$ become aligned with the eigenvectors corresponding to the leading $r_\star$ eigenvalues of $\mathcal{A}^*\mathcal{A}\left(XX^T\right)$. We observe in (6) that in the spectral phase increasing r, i.e. the amount of parameters, decreases the number of iterations in this phase. As we will explain in the supplementary, the reason is that

increasing r decreases the angle between the column space of the initialization U_0 and the span of the eigenvectors corresponding to the leading $r_\star$ eigenvalues of $\mathcal{A}^*\mathcal{A}\left(XX^T\right)$ used in spectral initialization. As a consequence, gradient descent needs fewer iterations to align these two subspaces.

In the saddle avoidance phase (Phase II), $\sigma_{r_\star}\left(U_t\right)$, the $r_\star$th largest singular value of U_t, grows geometrically until it is on the order of $\sigma_{\min}\left(X\right)$. Hence, this duration depends on the ratio between the $\sigma_{\min}\left(X\right)$ and the the scale of initialization α. This is clearly reflected in the upper bound on the number of needed iterations in equation (6).

In Phase III, the local refinement phase, the matrix $U_tU_t^T$ converges towards XX^T. In particular, at iteration $\hat{t}$ the test error obeys (5). We observe that a smaller α allows for a smaller test error in (5) but per (6) this higher accuracy is achieved with a modest increase in the required iterations.

4 A glimpse of our analysis

In our proofs, we show that the trajectory of the gradient descent iterations can be approximately decomposed into three phases: (I) a *spectral or alignment phase* where we show that gradient descent from random initialization behaves akin to spectral initialization allowing us to show that at the end of this phase the column spaces of the iterates U_t and the ground truth matrix X are sufficiently aligned, (II) a *saddle avoidance phase*, where we show that the trajectory of the gradient iterates move away from certain degenerate saddle points , and (III) a *refinement phase*, where the product of the gradient descent iterates $U_tU_t^T$ converges quickly to the underlying low-rank matrix XX^T. The latter result holds up to a small error that is commensurate with the scale of the initialization and tends to zero as the scale of the initialization goes to zero.

To formalize the above, we use L and L_t to denote the subspaces spanned by the eigenvectors corresponding to the $r_\star$ largest eigenvalues of the matrix $\mathcal{A}^*\mathcal{A}\left(XX^T\right)$, and $U_tU_t^T$, respectively. Moreover, for a subspace L of dimension $r_\star$ we use $V_L \in \mathbb{R}^{n\times r_\star}$ to denote an orthonormal matrix whose columns span the subspace L. Note that L is the subspace, which is obtained by commonly used spectral methods. Using this notation, Figure 2a depicts the three phases described above.

In the spectral phase we will prove that we can approximate the iterate U_t by

$$U_t \approx \underbrace{\left(\mathrm{Id} + \mu\mathcal{A}^*\mathcal{A}\left(XX^T\right)\right)^t}_{=:Z_t} U_0 = Z_1^t U_0 := \tilde{U}_t. \tag{7}$$

We note that the matrix $Z_1 = \mathrm{Id} + \mu\mathcal{A}^*\mathcal{A}\left(XX^T\right)$ is the basis for the commonly used spectral initialization, where typically a factorization of the rank r_* approximation of this matrix is used as the initialization [4, 3, 38]. Therefore, the approximation (7) suggests that gradient descent iterates modulo the normalization are akin to running power method on Z_1. Hence, we expect that at the end of the spectral phase the subspace L_t to be closely aligned with the subspace L, i.e. the subspace obtained by commonly used spectral initialization techniques. In particular, this also implies that L_t is also aligned with the subspace X. Figure 2b clearly illustrates that the first few iterations of gradient descent behave essentially identical to the power method, confirming our intuition.

The description of the second and third phase is more elaborate and technical in nature and we defer to the supplementary for a more detailed and intuitive explanation. However, to give a brief description, in these two phases we will decompose the iterates U_t into the sum of two matrices, a "signal" matrix of rank $r_\star$ and a "noise" matrix of rank at most $r - r_\star$. In Phase (II) we will prove that the smallest singular value of the signal term, which is approximately the same as $\sigma_{r_\star}\left(U_t\right)$ grows, whereas the spectral norm of the noise matrix grows at a much slower rate. We will also show that in this phase the columns of the signal term stay approximately aligned with the span of the matrix X. In Phase (III) we provide a local convergence argument, which shows that $U_tU_t^T$ converges towards XX^T.

5 Numerical experiments

In this section, we perform several numerical experiments to corroborate our theoretical results.

Experimental setup. For the experiments we set the ground truth matrix $X \in \mathbb{R}^{n\times r_\star}$ to be a random orthogonal matrix with $n = 200$ and $r_\star = 5$. Moreover, we use $m = 10nr_\star = 50n$ random Gaussian

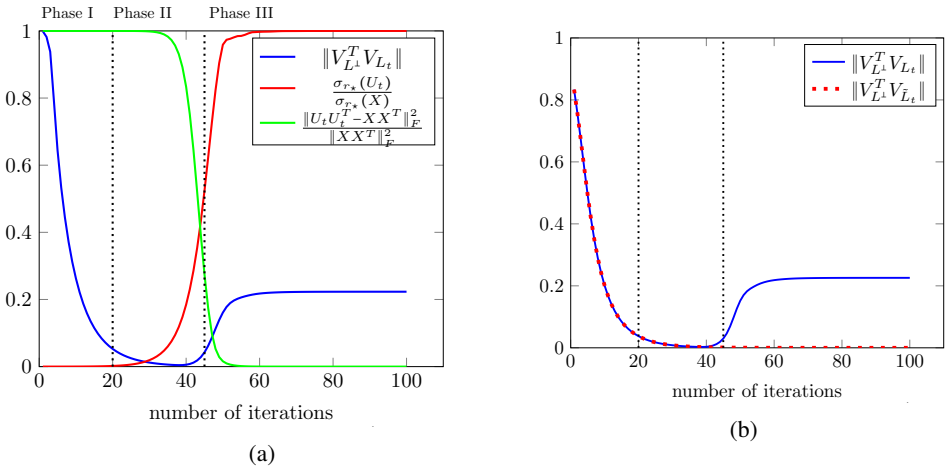


Figure 2: **(a) Depiction of the three phases of convergence.** This figure demonstrates that the convergence analysis can be divided into three phases: (I) spectral/alignment phase; (II) saddle avoidance phase and (III) the refinement phase. We see that in the first phase the first $r_\star$ eigenvectors of $U_t U_t^T$ rapidly learn the subspace corresponding to the first $r_\star$ eigenvectors of $\mathcal{A}^*\mathcal{A}\left(XX^T\right)$, i.e. the angle $\|V_{L^\perp}^T V_{L_t}\|$ becomes small. The $r_\star$th largest singular value of U_t is still small in this phase and the (normalized) test error $\|U_t U_t - XX^T\|_F^2 / \|XX^T\|_F^2$ has not decreased yet. In Phase (II), however, we see that $\sigma_{r_\star}(U_t)$ is growing, whereas the loss begins to decrease in this phase and the subspaces stay aligned. In Phase (III) we see that the test error is converging towards 0 rapidly, meaning that $U_t U_t^T$ converges to XX^T. Consequently, $\sigma_{r_\star}(U_t)/\sigma_{r_\star}(X)$ converges to 1 (red curve). We also see that in this phase the angle $\|V_{L^\perp}^T V_{L_t}\|$ grows again, until it reaches a certain threshold. This is because in this phase the top $r_\star$ eigenvalues of $U_t U_t^T$ become aligned with the eigenvectors of XX^T. **(b) Depiction of the spectral alignment phase: in the first few iterations, gradient descent with small initialization behaves like a power method.** Denote by $\tilde{L}_t$ the subspace spanned by the eigenvectors corresponding to the $r_\star$ largest eigenvalues of the matrix $\widetilde{U}_t \widetilde{U}_t^T$. Analogously as before denote by $V_{\tilde{L}_t}$ an orthonormal matrix, whose columns span the subspace $\tilde{L}_t$. In this figure, we observe that in the first iterations U_t and $\widetilde{U}_t$ learn the subspace L at almost exactly the same rate.

measurements. The initialization U is chosen as in Theorem 3.3 and we use a step size of $\mu = 1/4$ which is consistent with these theorems. We note that while all experimental depictions are based on a single trial, in line with the NeuRIPS guidelines we have drawn these curves multiple times (not depicted) and the behavior of the plots do not change.

Depiction of the three phases and the role of overparameterization. In our first experiment, we want to examine how increasing the number of parameters via increasing the number of the columns r of the matrix $U_t \in \mathbb{R}^{n \times r}$, affects the spectral phase. To this aim we set the scale of initialization to $\alpha = 1/\left(70n^2\right)$. Recall from Section 4 that L denotes the subspace spanned by the eigenvectors corresponding to the leading $r_\star$ singular values of $\mathcal{A}^*\mathcal{A}(XX^T)$ and L_t denotes the subspace spanned by the left-singular vectors corresponding to the largest $r_\star$ singular values of U_t.

Spectral phase and alignment under different levels of overparameterization. First, we examine how the angle between these two subspaces (i.e. $\|V_{L^\perp}^T V_{L_t}\|$) changes in the first few iterations. We depict the results for different r in Figure 3a. We see that in the first few iterations, i.e. in the spectral phase, this angle converges towards zero. This confirms the main conclusion of this paper that the first few iterations of gradient descent from small random initialization indeed behaves akin to running power method for spectral initialization. This experiment also shows that changing the number of columns r of U_t has an interesting effect on the spectral phase. In particular, increasing r allows the gradient descent algorithm to learn the subspace L with fewer iterations, i.e. $\|V_{L^\perp}^T V_{L_t}\|$ becomes small with fewer iterations. This is in accordance with our theory for $r_\star \leq r \leq n$ (see, for example, the first

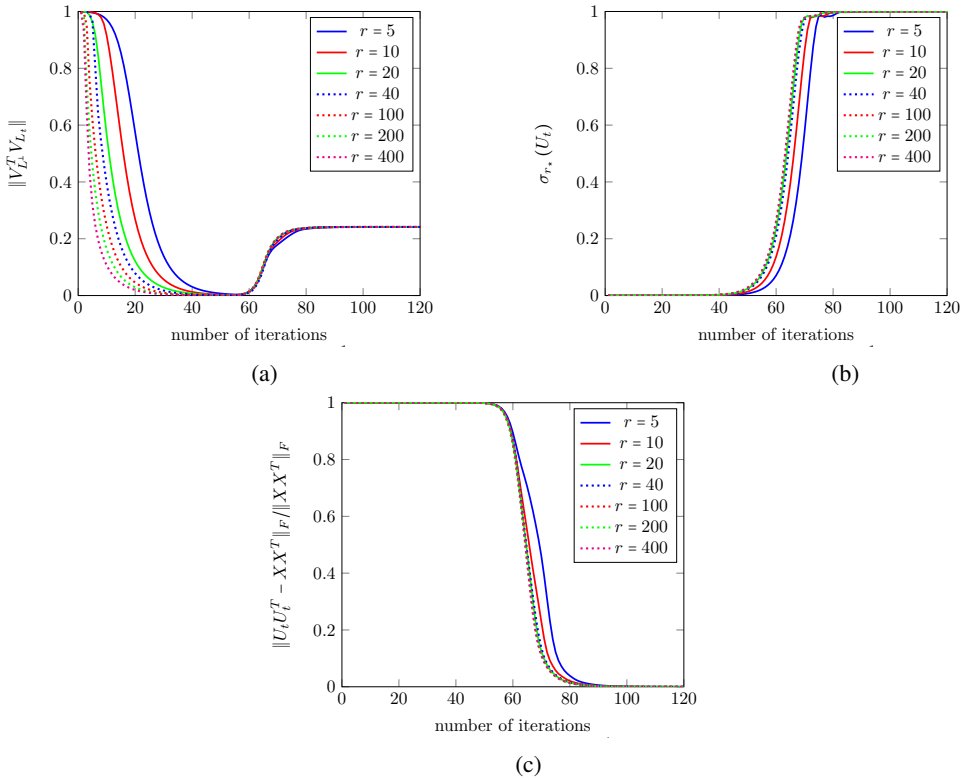


Figure 3: Impact of different levels of overparameterization in the spectral phase on (a) the angle $\|V_{L^\perp}^T V_{L_t}\|$ and (b) the $r_\star$th largest singular value, (c) the trajectory of the (normalized) test error $\|U_t U_t^T - XX^T\|_F / \|XX^T\|_F$.

summand on the right-hand side of equation (6)), where we show that more overparameterization allows gradient descent to leave the spectral phase earlier. Interestingly, this improvement continues to hold even when increasing r beyond n allowing for even faster convergence of $\|V_{L^\perp}^T V_{L_t}\|$. This holds even though in this case the rank of U_0 is still not larger than n. One potential explanation for this phenomenon might be that for such a choice of r the matrix U_0 is better conditioned.

Growth of $\sigma_{r_\star}(U_t)$ and saddle avoidance. In Figure 3b we depict how $\sigma_{r_\star}(U_t)$ grows during the training for different choices of r. We see that the curves look similar, although for smaller r the growth phase sets in at a slightly later time. This is due to the fact that for smaller r, as we have seen in Figure 3a, Phase I, the spectral phase takes longer to complete.

Evolution of the test error and the refinement phase. Similarly, in Figure 3c we depict how the (normalized) test error $\|U_t U_t^T - XX^T\|_F / \|XX^T\|_F$ evolves during the training for different choices of r. We observe that for smaller r the third phase sets in slightly later. Again, this is due to the fact that for smaller r the spectral phase takes slightly longer to complete.

Test error under different scales of initialization. In the next experiment, we focus on understanding how the scale of initialization α affects the generalization error $\|U_t U_t^T - XX^T\|_F^2$. For that, we set $r = 180$ and run gradient descent with for different choices of α. We stop as soon as the training error becomes small ($f(U_t) \leq 0.5 \cdot 10^{-9}$). We depict the results in Figure 4. We see that the test error decreases as α decreases. In particular, this figure indicates that the test error depends polynomially on the scale of initialization α. This is in line with our theory, where we also show that the test error decreases at least with the rate $\alpha^{21/16}$ (see inequality (5) in Theorem 3.3).

Change of test and train error during the training. In the next experiment, we examine how the test error $\|U_t U_t^T - XX^T\|_F^2$ and the train error $f(U_t)$ changes throughout training and, in particular, how this depends on the scale of initialization. To this aim, we run gradient descent with $4 \cdot 10^5$ iterations. We see that for a small scale of initialization, $\alpha = 10^{-3}$, which is the scenario studied in

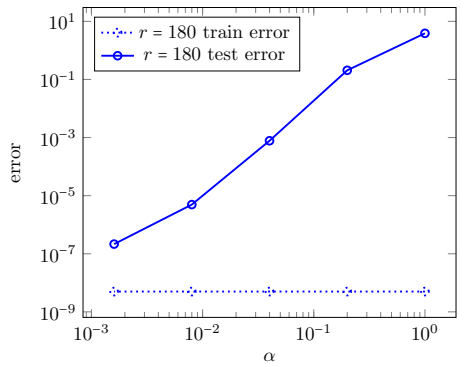
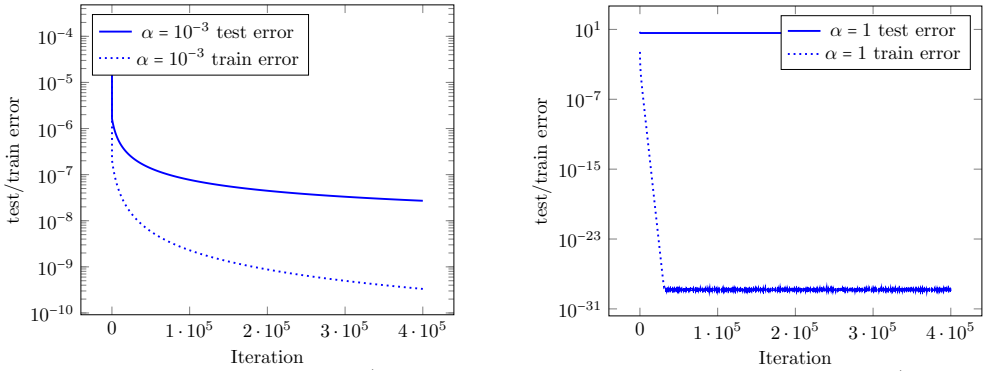


Figure 4: Relativ test error $\frac{\|U_t U_t^T - XX^T\|_F}{\|XX^T\|_F}$ for different scales of initialization α .



(a) $\alpha = 10^{-3}$ (b) $\alpha = 1$

Figure 5: Change of test error and train error for (a) small and (b) large α during training.

this paper, both test error and train error decrease throughout the training. In particular, the test error is still decreasing even when the train error is already small. For large scale of initialization $\alpha = 1$, we observe a very different behaviour. We see that the train error converges with linear rate until machine precision is reached. However, the test error barely changes throughout the training. This scale of initialization corresponds to the *lazy training regime* or *NTK-regime*, where the parameters stay close to the initialization during the training.

Conclusion and Broader Impacts

In this paper we focused on demystifying the role of initialization when training overparameterized models by showing that small random initialization followed by a few iterations of gradient descent behaves akin to popular spectral methods. We also show that this *implicit spectral bias* from small random initialization, which is provably more prominent for overparameterized models, also puts the gradient descent iterations on a particular trajectory towards solutions that are not only globally optimal but also generalize well. While in this paper our main focus was on low-rank matrix reconstruction, we believe that our analysis holds more generally for a variety of contemporary overparameterized machine learning and signal estimation tasks including neural network training. This is a tantalizing future research direction. Despite being theoretical/foundational in nature our results have potential for broader practical impact. In particular, low rank reconstruction problems are an important component of many recommender engines and our insights may guide better algorithm and systems designs for such engines. More broadly, training overparameterized models using stochastic GD starting from small random initialization is the work-horse of modern learning including deep learning and our insights may in the long term help enable more efficient/reliable training with a smaller carbon footprint and improved test accuracy. As with other technologies such insights may potentially also be used nefariously.

References

[1] Emmanuel J. Candès, Xiaodong Li, and Mahdi Soltanolkotabi. Phase retrieval via Wirtinger flow: theory and algorithms. *IEEE Trans. Inf. Theory*, 61(4):1985–2007, 2015.

[2] Yuxin Chen and Emmanuel J. Candès. Solving random quadratic systems of equations is nearly as easy as solving linear systems. *Commun. Pure Appl. Math.*, 70(5):822–883, 2017.

[3] Cong Ma, Kaizheng Wang, Yuejie Chi, and Yuxin Chen. Implicit regularization in nonconvex statistical estimation: gradient descent converges linearly for phase retrieval, matrix completion, and blind deconvolution. *Found. Comput. Math.*, 20(3):451–632, 2020.

[4] S. Tu, R. Boczar, M. Simchowitz, M. Soltanolkotabi, and B. Recht. Low-rank solutions of linear matrix equations via procrustes flow. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning (ICML)*, volume 48, pages 964–973. Journal of Machine Learning Research, 2016.

[5] Xiaodong Li, Shuyang Ling, Thomas Strohmer, and Ke Wei. Rapid, robust, and reliable blind deconvolution via nonconvex optimization. *Appl. Comput. Harmon. Anal.*, 47(3):893–934, 2019.

[6] Shuyang Ling and Thomas Strohmer. Regularized gradient descent: a non-convex recipe for fast joint blind deconvolution and demixing. *Inf. Inference*, 8(1):1–49, 2019.

[7] Praneeth Netrapalli, Prateek Jain, and Sujay Sanghavi. Phase retrieval using alternating minimization. *IEEE Trans. Signal Process.*, 63(18):4814–4826, 2015.

[8] Irène Waldspurger. Phase retrieval with random Gaussian sensing vectors by alternating projections. *IEEE Trans. Inf. Theory*, 64(5):3301–3312, 2018.

[9] Ju Sun, Qing Qu, and John Wright. When are nonconvex problems not scary? *arXiv preprint arXiv:1510.06096*, 2015.

[10] Yurii Nesterov and B. T. Polyak. Cubic regularization of Newton method and its global performance. *Math. Program.*, 108(1 (A)):177–205, 2006.

[11] Jorge Nocedal and Stephen J. Wright. Trust-region methods. *Numerical Optimization*, pages 66–100, 2006.

[12] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M. Kakade, and Michael I. Jordan. How to escape saddle points efficiently. page 1724–1732, 2017.

[13] Rong Ge, Furong Huang, Chi Jin, and Yang. Yuan. Escaping from saddle points: online stochastic gradient for tensor decomposition. In *Proceedings of The 28th Conference on Learning Theory*, pages 797–842, 2015.

[14] Maxim Raginsky, Alexander Rakhlin, and Matus Telgarsky. Non-convex learning via stochastic gradient langevin dynamics: a nonasymptotic analysis. pages 1674–1703, 2017.

[15] Yuchen Zhang, Percy Liang, and Moses Charikar. A hitting time analysis of stochastic gradient langevin dynamics. In Satyen Kale and Ohad Shamir, editors, *Proceedings of the 2017 Conference on Learning Theory*, volume 65 of *Proceedings of Machine Learning Research*, pages 1980–2022. PMLR, 07–10 Jul 2017.

[16] Jason D. Lee, Max Simchowitz, Michael I. Jordan, and Benjamin Recht. Gradient descent only converges to minimizers. In Vitaly Feldman, Alexander Rakhlin, and Ohad Shamir, editors, *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 1246–1257, Columbia University, New York, New York, USA, 23–26 Jun 2016. PMLR.

[17] Simon S Du, Chi Jin, Jason D Lee, Michael I Jordan, Aarti Singh, and Barnabas Poczos. Gradient descent can take exponential time to escape saddle points. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[18] Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. In *Advances in Neural Information Processing Systems*, pages 7413–7424, 2019.

[19] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pages 8571–8580, 2018.

[20] Samet Oymak and Mahdi Soltanolkotabi. Overparameterized nonlinear learning: Gradient descent takes the shortest path? In *International Conference on Machine Learning*, pages 4951–4960. PMLR, 2019.

[21] Samet Oymak and Mahdi Soltanolkotabi. Towards moderate overparameterization: global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 2020.

[22] Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pages 1675–1685. PMLR, 2019.

[23] Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *36th International Conference on Machine Learning, ICML 2019*, pages 477–502. International Machine Learning Society (IMLS), 2019.

[24] Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671, 2018.

[25] Lénaïc Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, pages 3036–3046. Curran Associates, Inc., 2018.

[26] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit. In *Conference on Learning Theory*, pages 2388–2464. PMLR, 2019.

[27] Adel Javanmard, Marco Mondelli, Andrea Montanari, et al. Analysis of a two-layer neural network via displacement convexity. *Ann. Statist.*, 48(6):3619–3642, 2020.

[28] Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of neural networks: a central limit theorem. *Stochastic Processes Appl.*, 130(3):1820–1852, 2020.

[29] Lénaïc Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. *Advances in Neural Information Processing Systems*, 32:2937–2947, 2019.

[30] Emmanuel J. Candès and Yaniv Plan. Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Trans. Inf. Theory*, 57(4):2342–2359, 2011.

[31] Martin Slawski, Ping Li, and Matthias Hein. Regularization-free estimation in trace regression with symmetric positive semidefinite matrices. *Advances in Neural Information Processing Systems*, 28:2782–2790, 2015.

[32] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

[33] Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. Global optimality of local search for low rank matrix recovery. 29, 2016.

[34] Richard Y. Zhang, Somayeh Sojoudi, and Javad Lavaei. Sharp restricted isometry bounds for the inexistence of spurious local minima in nonconvex matrix recovery. *J. Mach. Learn. Res.*, 20(114):1–34, 2019.

[35] Yuxin Chen, Yuejie Chi, Jianqing Fan, and Cong Ma. Gradient descent with random initialization: fast global convergence for nonconvex phase retrieval. *Math. Program.*, 176(1-2 (B)):5–37, 2019.

[36] Yuanzhi Li, Tengyu Ma, and Hongyang Zhang. Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. volume 75 of *Proceedings of Machine Learning Research*, pages 2–47. PMLR, 06–09 Jul 2018.

[37] Stephen Tu, Ross Boczar, Max Simchowitz, Mahdi Soltanolkotabi, and Ben Recht. Low-rank solutions of linear matrix equations via procrustes flow. In *International Conference on Machine Learning*, pages 964–973. PMLR, 2016.

[38] Ji Chen, Dekai Liu, and Xiaodong Li. Nonconvex rectangular matrix completion via gradient descent without ℓ_2, ∞ regularization. *IEEE Trans. Inf. Theory*, 66(9):5806–5841, 2020.

[39] Ju Sun, Qing Qu, and John Wright. A geometric analysis of phase retrieval. *Found. Comput. Math.*, 18(5):1131–1198, 2018.

[40] Rong Ge, Jason D Lee, and Tengyu Ma. Matrix completion has no spurious local minimum. *Advances in Neural Information Processing Systems*, 29:2973–2981, 2016.

[41] Richard Y. Zhang, Somayeh Sojoudi, and Javad Lavaei. Sharp restricted isometry bounds for the inexistence of spurious local minima in nonconvex matrix recovery. *J. Mach. Learn. Res.*, 20(114):1–34, 2019.

[42] Suriya Gunasekar, Blake Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nathan Srebro. Implicit regularization in matrix factorization. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6152–6160, 2017.

[43] Noam Razin and Nadav Cohen. Implicit regularization in deep learning may not be explainable by norms. *arXiv preprint arXiv:2005.06398*, 2020.

[44] Zhiyuan Li, Yuping Luo, and Kaifeng Lyu. Towards resolving the implicit bias of gradient descent for matrix factorization: Greedy low-rank learning. In *International Conference on Learning Representations*, 2021.

[45] Behrooz Ghorbani, Song Mei, Theodor Misiakiewicz, and Andrea Montanari. When do neural networks outperform kernel methods? *arXiv preprint arXiv:2006.13409*, 2020.

[46] Xiang Wang, Chenwei Wu, Jason D Lee, Tengyu Ma, and Rong Ge. Beyond lazy training for over-parameterized tensor decomposition. *Advances in Neural Information Processing Systems*, 33, 2020.

[47] Rong Ge, Jason D. Lee, and Tengyu Ma. Learning one-hidden-layer neural networks with landscape design. In *International Conference on Learning Representations*, 2018.

[48] Yuanzhi Li, Tengyu Ma, and Hongyang R Zhang. Learning over-parametrized two-layer relu neural networks beyond ntk. *arXiv preprint arXiv:2007.04596*, 2020.

[49] Peter Bartlett, Dave Helmbold, and Philip Long. Gradient descent with identity initialization efficiently learns positive definite linear transformations by deep residual networks. 80:521–530, 10–15 Jul 2018.

[50] Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. In *International Conference on Machine Learning*, pages 244–253. PMLR, 2018.

[51] Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu. A convergence analysis of gradient descent for deep linear neural networks. *arXiv preprint arXiv:1810.02281*, 2018.

[52] Bubacarr Bah, Holger Rauhut, Ulrich Terstiege, and Michael Westdickenberg. Learning deep linear neural networks: Riemannian gradient flows and convergence to global minimizers. *arXiv preprint arXiv:1910.05505*, 2019.

[53] Hung-Hsu Chou, Carsten Gieshoff, Johannes Maly, and Holger Rauhut. Gradient descent for deep matrix factorization: Dynamics and implicit bias towards low rank. *arXiv preprint arXiv:2011.13772*, 2020.

[54] Suriya Gunasekar, Jason D Lee, Daniel Soudry, and Nati Srebro. Implicit bias of gradient descent on linear convolutional networks. 31, 2018.

[55] Yuxin Chen, Yuejie Chi, Jianqing Fan, and Cong Ma. Spectral methods for data science: A statistical perspective. *arXiv preprint arXiv:2012.08496*, 2020.

[56] Mark Rudelson and Roman Vershynin. Smallest singular value of a random rectangular matrix. *Commun. Pure Appl. Math.*, 62(12):1707–1739, 2009.

[57] Simon Foucart and Holger Rauhut. *A mathematical introduction to compressive sensing*. New York, NY: Birkhäuser/Springer, 2013.

[58] Per-Ake Wedin. Perturbation bounds in connection with singular value decomposition. *BIT, Nord. Tidskr. Inf.-behandl.*, 12:99–111, 1972.

[59] Chandler Davis and W. M. Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM J. Numer. Anal.*, 7:1–46, 1970.

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] See Section 3, where we discuss the suboptimal dependence of the sample complexity with respect to $r_\star$
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section "Conclusion and Broader Impacts"
   (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] See Section 3.
   (b) Did you include complete proofs of all theoretical results? [Yes] See supplemental 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] See supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 5.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] See the dscussion in Section 5, Experimental setup.
   (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)? [No] The experiments are computationally rather inexpensive and can be run on a personal computer.
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? [N/A]
   (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]