# Preconditioned Gradient Descent for Over-Parameterized Nonconvex Matrix Factorization

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

In practical instances of nonconvex matrix factorization, the rank of the true solution $r^\star$ is often unknown, so the rank $r$ of the model can be overspecified as $r > r^\star$. This over-parameterized regime of matrix factorization significantly slows down the convergence of local search algorithms, from a linear rate with $r = r^\star$ to a sublinear rate when $r > r^\star$. We propose an inexpensive preconditioner for the matrix sensing variant of nonconvex matrix factorization that restores the convergence rate of gradient descent back to linear, even in the over-parameterized case, while also making it agnostic to possible ill-conditioning in the ground truth. Classical gradient descent in a neighborhood of the solution slows down due to the need for the model matrix factor to become singular. Our key result is that this singularity can be corrected by $\ell_2$ regularization with a specific range of values for the damping parameter. In fact, a good damping parameter can be inexpensively estimated from the current iterate. The resulting algorithm, which we call preconditioned gradient descent or PrecGD, is stable under noise, and converges linearly to an information theoretically optimal error bound. Our numerical experiments find that PrecGD works equally well in restoring the linear convergence of other variants of nonconvex matrix factorization in the over-parameterized regime.

## 1 Introduction

Numerous problems in machine learning can be reduced to the *matrix factorization* problem of recovering a low-rank positive semidefinite matrix $M^\star \succeq 0$, given a small number of potentially noisy observations [47, 30, 9, 12, 1, 29, 38]. In every case, the most common approach is to formulate an $n \times n$ candidate matrix $M = XX^T$ in factored form, and to minimize a *nonconvex* empirical loss $f(X)$ over its $n \times r$ low-rank factor $X$. But in most real applications of nonconvex matrix factorization, the rank of the ground truth $r^\star = \mathrm{rank}(M^\star)$ is unknown. It is reasonable to choose the rank $r$ of the model $XX^T$ conservatively, setting it to be potentially larger than $r^\star$, given that the ground truth can be exactly recovered so long as $r \geq r^\star$. In practice, this will often lead to an *over-parameterized* regime, in which $r > r^\star$, and we have specified more degrees of freedom in our model $XX^T$ than exists in the underlying ground truth $M^\star$.

Zhuo et al. [54] recently pointed out that nonconvex matrix factorization becomes substantially less efficient in the over-parameterized regime. For the prototypical instance of matrix factorization known as *matrix sensing* (see Section 3 below for details) it is well-known that, if $r = r^\star$, then (classic) gradient descent or GD

$$X_{k+1} = X_k - \alpha \nabla f(X_k) \tag{GD}$$

converges at a linear rate, to an $\epsilon$-accurate iterate in $k = O(\log(1/\epsilon))$ iterations [51, 44]. But in the case that $r > r^\star$, Zhuo et al. [54] proved that gradient descent slows down to a *sublinear* convergence

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

rate, now requiring $k = \mathrm{poly}(1/\epsilon)$ iterations to yield a comparable $\epsilon$-accurate solution. This is a dramatic, exponential slow-down: whereas 10 digits of accuracy can be expected in a just few hundred iterations when $r = r^\star$, tens of thousands of iterations might produce just 1-2 accurate digits once $r > r^\star$. The slow-down occurs even if $r$ is just off by one, as in $r = r^\star + 1$.

As the true rank $r^\star$ is generally unknown, this recent result suggests that most practical implementations of gradient descent do not actually enjoy the rapid convergence that is guaranteed by theory if $r^\star$ were known. In this paper, we present an inexpensive *preconditioner* for gradient descent. We prove, for the matrix sensing variant of nonconvex matrix factorization, that the preconditioner restores the convergence rate of gradient descent back to linear, even in the over-parameterized case, while also making it agnostic to possible ill-conditioning in the ground truth. The resulting algorithm, which we call preconditioned gradient descent or PrecGD, maintains a similar per-iteration cost to regular gradient descent, is stable under noise, and converges linearly to an information theoretically optimal error bound.

We also perform numerical experiments on other variants of nonconvex matrix factorization, with different choices of the empirical loss function $f$. In particular, we consider different $\ell_p$ norms with $1 \leq p < 2$, in order to gauge the effectiveness of PrecGD for increasing nonsmooth loss functions. Our numerical experiments find that, if regular gradient descent is capable of converging quickly when the rank is known $r = r^\star$, then PrecGD restores this rapid converging behavior when $r > r^\star$. PrecGD is able to overcome ill-conditioning in the ground truth, and converge reliably without exhibiting sporadic behavior.

## 2 Proposed Algorithm: Preconditioned Gradient Descent

Our preconditioner is inspired by a recent work of Tong et al. [43] on matrix sensing with an ill-conditioned ground truth $M^\star$. Over-parameterization can be viewed as the limit of this regime, in which $\lambda_r(M^\star)$, the $r$-th largest eigenvalue of $M^\star$, is allowed to approach all the way to zero. For finite but potentially very small values of $\lambda_r(M^\star) > 0$, Tong et al. [43] suggests the following iterations, which they named *scaled* gradient descent or ScaledGD:

$$X_{k+1} = X_k - \alpha \nabla f(X_k)(X_k^T X_k)^{-1}. \tag{ScaledGD}$$

They prove that the scaling allows the iteration to make a large, constant amount of progress at every iteration, independent of the value of $\lambda_r(M^\star) > 0$. However, applying this same scheme to the over-parameterized case with $\lambda_r(M^\star) = 0$ results in an inconsistent, sporadic behavior.

The issues encountered by both regular GD and ScaledGD with over-parameterization $r > r^\star$ can be explained by the fact that our iterate $X_k$ must necessarily become *singular* as our rank-$r$ model $X_k X_k^T$ converges towards the rank-$r^\star$ ground truth $M^\star$. For GD, this singularity causes the per-iteration progress itself to decay, so that more and more iterations are required for each fixed amount of progress. ScaledGD corrects for this decay in per-iteration progress by suitably rescaling the search direction. However, the rescaling itself requires inverting a near-singular matrix, which causes algorithm to take on sporadic values.

A classical remedy to issues posed by singular matrices is $\ell_2$ regularization, in which the singular matrix is made "less singular" by adding a small identity perturbation. Applying this idea to ScaledGD yields the following iterations

$$X_{k+1} = X_k - \alpha \nabla f(X_k)(X_k^T X_k + \eta_k I_r)^{-1}, \tag{PrecGD}$$

where $\eta_k \geq 0$ is the *damping* parameter specific to the $k$-th iteration. There are several interpretations to this scheme, but the most helpful is to view $\eta$ as a parameter that allows us to interpolate between ScaledGD (with $\eta = 0$) and regular GD (in the limit $\eta \to \infty$). In this paper, we prove for matrix sensing that, if the $k$-th damping parameter $\eta_k$ is chosen within a constant factor of the error

$$C_{\mathrm{lb}} \|X_k X_k^T - M^\star\|_F \leq \eta_k \leq C_{\mathrm{ub}} \|X_k X_k^T - M^\star\|_F, \quad \text{where } C_{\mathrm{lb}}, C_{\mathrm{ub}} > 0 \text{ are abs. const.} \tag{1}$$

then the resulting iterations are guaranteed to converge linearly, at a rate that is independent of both over-parameterization and ill-conditioning in the ground truth $M^\star$. With noisy measurements, setting $\eta_k$ to satisfy (1) will allow the iterations to converge to an error bound that is well-known to be minimax optimal up to logarithmic factors [6].

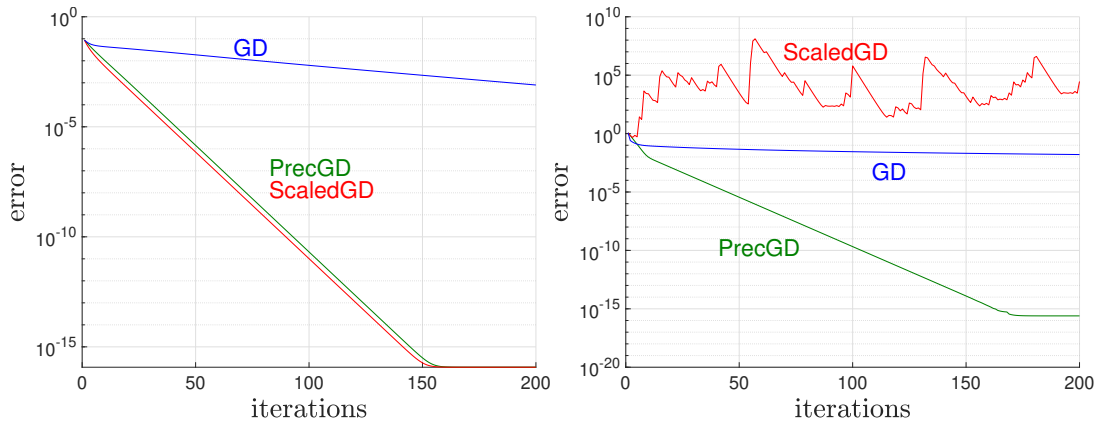


Figure 1: **PrecGD converges linearly in the overparameterized regime.** Convergence of regular gradient descent (GD), ScaledGD and PrecGD for noiseless matrix sensing (with data taken from [48, 50]) from the same initial points and using the same learning rate $\alpha = 5 \times 10^{-2}$. **(Left** $r = r^*$**)** Set $n = 3$ and $r^* = r = 2$. All three methods convergence at a linear rate, though GD converges at a slower rate due to ill-conditioning in the ground truth. **(Right** $r > r^*$**)** With $n = 2$, $r = 2$ and $r^* = 1$, over-parameterization causes gradient descent to slow down to a sublinear rate. ScaledGD also behaves sporadically. Only PrecGD converges linearly to the ground truth.

We refer to the resulting iterations (with a properly chosen $\eta_k$) as *preconditioned* gradient descent, or PrecGD for short. For matrix sensing with noiseless measurements, an optimal $\eta_k$ that satisfies the condition (1) is obtained for free by setting $\eta_k = \sqrt{f(X_k)}$. In the case of noisy measurements, we show that a good choice of $\eta_k$ is available based on an approximation of the noise variance.

## 3 Background and Related Work

**Notations.** We use $\|\cdot\|_F$ to denote the Frobenius norm of a matrix and $\langle \cdot, \cdot \rangle$ is the corresponding inner product. We use $\gtrsim$ to denote an inequality that hides a constant factor. The big-O notation $\tilde{O}$ hides logarithimic factors. The gradient of the objective is denoted by $\nabla f(X) \in \mathbb{R}^{n \times r}$. The eigenvalues are assumed to be in decreasing order:$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_r$.

The symmetric, linear variant of matrix factorization known as matrix sensing aims to recover a positive semidefinite, rank-$r^\star$ ground truth matrix $M^\star$, from a small number $m$ of possibly noisy measurements

$$y = \mathcal{A}(M^\star) + \varepsilon, \qquad \text{where } \mathcal{A}(M^\star) = [\langle A_1, M^\star \rangle, \langle A_2, M^\star \rangle, \ldots, \langle A_m, M^\star \rangle]^T,$$

in which $\mathcal{A}$ is a linear measurement operator, and the length-$m$ vector $\epsilon$ models the unknown measurement noise. A distinguishing feature of matrix sensing is that $\mathcal{A}$ is assumed to satisfy the *restricted isometry property* [37, 6]. Throughout this paper, we will always assume that $\mathcal{A}$ satisfies RIP with parameters $(2r, \delta)$.

**Definition 1** (RIP)**.** The linear operator $\mathcal{A}$ satisfies RIP with parameters $(2r, \delta)$ if there exists constants $0 \leq \delta < 1$ and $m > 0$ such that, for every rank-$2r$ matrix $M$, we have

$$(1 - \delta)\|M\|_F^2 \leq \frac{1}{m}\|\mathcal{A}(M)\|^2 \leq (1 + \delta)\|M\|_F^2.$$

The most common approach for matrix sensing is to use a simple algorithm like gradient descent to minimize the *nonconvex* loss function:

$$f(X) = \frac{1}{m}\left\|y - \mathcal{A}(XX^T)\right\| = \frac{1}{m}\left\|\mathcal{A}(M^\star - XX^T) + \epsilon\right\|^2. \tag{2}$$

Recent work has provided a theoretical explanation for the empirical success of this nonconvex approach. Two lines of work have emerged.

**Local Guarantees.** One line of work studies gradient descent initialized inside a neighborhood of the ground truth where $X_0 X_0^T \approx M^\star$ already holds [44, 52, 3, 10, 31]. Such an initial point can be found

using spectral initialization, see also [26, 10, 14, 42, 34]. With exact rank $r = r^\star$, previous authors showed that gradient descent converges at a linear rate [51, 44]. In the over-parameterized regime, however, local restricted convexity no longer holds, so the linear convergence rate is lost. Zhuo et al. [54] showed that while spectral initialization continues to work under over-parameterization, gradient descent now slows down to a sublinear rate, but it still converges to a statistical error bound of $\tilde{\mathcal{O}}(\sigma^2 n r^\star/m)$, which is known to be minimax optimal up to logarithmic factors [6]. In this paper, we prove that PrecGD with a damping parameter $\eta_k$ satisfying (1) also converges to an $\tilde{\mathcal{O}}(\sigma^2 n r^\star/m)$ statistical error bound.

**Global Guarantees.** A separate line of work [4, 27, 41, 19, 20, 13, 40, 50, 49] established global properties of the landscapes of the nonconvex objective $f$ in (2) and its variants and showed that local search methods can converge globally. With exact rank $r = r^\star$, Bhojanapalli et al. [4] proved that $f$ has no spurious local minima, and that all saddles points have a strictly negative descent direction (strict saddle property [18], see also [25, 20]). In the over-parameterized regime, however, the strict saddle property is lost. While the problem continues to have no spurious local minima [49], we are no longer guaranteed to recover the ground truth in polynomial time.

**Other related work.** Here we mention some other techniques can be use to solve matrix sensing in the over-parameterized regime. Classically, matrix factorization was solved via its convex SDP relaxation [37, 6, 32, 7, 8]. The resulting $\mathcal{O}(n^3)$ to $\mathcal{O}(n^6)$ time complexity [2] limits this technique to smaller problems, but these guarantees hold without prior knowledge on the true rank $r^\star$. First-order methods, such as ADMM [45, 36, 53] and soft-thresholding [5], can be used to solve these convex problems with a per-iteration complexity comparable to nonconvex gradient descent, but they likewise suffer from a sublinear convergence rate. Local recovery via spectral initialization was originally proposed for alternating minimization and other projection techniques [24, 23, 32, 34, 14, 46, 39]. These also continue to work, though a drawback here is a higher per-iteration cost when compared to simple gradient methods. Finally, we mention a recent result of Li et al. [28], which showed in the over-parameterized regime that gradient descent with early termination enjoys an algorithmic regularization effect.

## 4 Sublinear convergence of Gradient Descent

In order to understand how to improve gradient descent in the over-parameterized regime, we must first understand why existing methods fail. For an algorithm that moves in a search direction $D$ with step-size $\alpha$, it is a standard technique to measure the corresponding decrement in $f$ with a Taylor-like expansion

$$f(X - \alpha D) \leq f(X) - \alpha \underbrace{\langle \nabla f(X), D \rangle}_{\text{linear progress}} + \alpha^2 \underbrace{(L/2)\|D\|_F^2}_{\text{inverse step-size}} \tag{3}$$

in which $L$ is the usual gradient Lipschitz constant (see e.g. Nocedal and Wright [35, Chapter 3]). A good search direction $D$ is one that maximizes the linear progress $\langle \nabla f(X), D \rangle$ while also keeping the inverse step-size $(L/2)\|D\|_F^2$ sufficiently small in order to allow a reasonably large step to be taken. As we will show in this section, the main issue with gradient descent in the over-parameterized regime is the first term, namely, that the linear progress goes down to zero as the algorithm makes progress towards the solution.

Classical gradient descent uses the search direction $D = \nabla f(X)$. Here, a common technique is to bound the linear progress at each iteration by a condition known as *gradient dominance* (or the Polyak-Łojasiewicz or PL inequality), which is written as

$$\langle \nabla f(X), D \rangle = \|\nabla f(X)\|_F^2 \geq \mu(f(X) - f^\star) \quad \text{where } \mu > 0 \text{ and } f^\star = \min f(X). \tag{4}$$

Substituting the inequality (4) into the Taylor-like expansion (3) leads to

$$\begin{aligned} f(X - \alpha D) &\leq f(X) - \alpha\|\nabla f(X)\|_F^2 + \alpha^2 (L/2)\|\nabla f(X)\|_F^2 \\ f(X - \alpha D) - f^\star &\leq [1 - \mu\alpha(1 - \alpha L/2)] \cdot (f(X) - f^\star). \end{aligned} \tag{5}$$

Here, we can always pick a small enough step-size $\alpha$ to guarantee linear convergence:

$$Q = 1 - \mu\alpha + \mu\alpha^2 L/2 < 1 \implies f(X_k) - f^\star \leq Q^k [f(X_0) - f^\star]. \tag{6}$$

In particular, picking the optimal step-size $\alpha = 1/L$ minimizes the convergence quotient $Q = 1 - 1/(2\kappa)$, where $\kappa = L/\mu$ is the usual *condition number*. This shows that, with an optimal step-size, gradient descent needs at most $O(\kappa \log(1/\epsilon))$ iterations to find an $\epsilon$-suboptimal $X$.

Matrix sensing with exact rank $r = r^\star$ is easily shown to satisfy gradient dominance (4) by manipulating existing results on (restricted) local strong convexity. In the over-parameterized case $r > r^\star$, however, local strong convexity is lost, and gradient dominance can fail to hold. Indeed, consider the following instance of matrix sensing, with true rank $r^\star = 1$, search rank $r = 2$, and $\mathcal{A}$ set to the identity

$$f(X) = \|XX^T - zz^T\|_F^2 \text{ where } X = \begin{bmatrix} 1 & 0 \\ 0 & \xi \end{bmatrix} \text{ and } z = \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \tag{7}$$

We can verify that $\|\nabla f(X)\|^2 = 4\xi^2[f(X) - f^\star]$, and this suggests that $f$ satisfies gradient dominance (4) with a constant of $\mu \leq 2\xi^2$. But $\xi$ is itself a variable that goes to zero as the candidate $XX^T$ approaches to ground truth $zz^T$. For every fixed $\mu > 0$ in the gradient dominance condition (4), we can find a counterexample $X$ in (7) with $\xi < \sqrt{\mu}/2$. Therefore, we must conclude that gradient dominance fails to hold, because the inequality in (4) can only hold for $\mu = 0$.

In fact, this same example also shows why classical gradient descent slows down to a sublinear rate. Applying gradient descent $X_{k+1} = X_k - \alpha\nabla f(X_k)$ with fixed step-size $\alpha$ to (7) yields a sequence of iterates of the same form

$$X_0 = \begin{bmatrix} 1 & 0 \\ 0 & \xi_0 \end{bmatrix}, \qquad X_{k+1} = \begin{bmatrix} 1 & 0 \\ 0 & \xi_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \xi_k - \alpha\xi_k^3 \end{bmatrix},$$

from which we can verify that $f(X_{k+1}) = [1 - 4\alpha\xi_k(1 + \alpha\xi_k)]^2 f(X_k)$. As each $k$-th $X_k X_k^T$ approaches $zz^T$, the element $\xi_k$ converges towards zero, and the convergence quotient $Q = [1 - 4\alpha\xi_k(1 + \alpha\xi_k)]^2$ approaches 1. We see a process of diminishing returns: every improvement to $f$ worsens the quotient $Q$, thereby reducing the progress achievable in the subsequent step. This is precisely the notion that characterizes sublinear convergence.

## 5 Linear Convergence for the Noiseless Case

To understand how it is possible make gradient descent converge linearly in the over-parameterized regime, we begin by considering gradient method under a *change of metric*. Let $\mathbf{P}$ be a real symmetric, positive definite $nr \times nr$ matrix. We define a corresponding $P$-inner product, $P$-norm, and dual $P$-norm on $\mathbb{R}^{n\times r}$ as follows

$$\langle X, Y\rangle_P \stackrel{\text{def}}{=} \text{vec}(X)^T\mathbf{P}\text{vec}(Y), \quad \|X\|_P \stackrel{\text{def}}{=} \sqrt{\langle X, X\rangle_P}, \quad \|X\|_{P*} \stackrel{\text{def}}{=} \sqrt{\text{vec}(X)^T\mathbf{P}^{-1}\text{vec}(X)},$$

where $\text{vec} : \mathbb{R}^{n\times r} \to \mathbb{R}^{nr}$ is the usual column-stacking operation. Consider descending in the direction $D$ satisfying $\text{vec}(D) = \mathbf{P}^{-1}\text{vec}(\nabla f(X))$; the resulting decrement in $f$ can be quantified by a $P$-norm analog of the Taylor-like expansion (3)

$$f(X - \alpha D) \leq f(X) - \alpha\langle\nabla f(X), D\rangle + \alpha^2(L_P/2)\|D\|_P^2 \tag{8}$$

$$= f(X) - \alpha(1 - \alpha(L_P/2))\|\nabla f(X)\|_{P*}^2 \tag{9}$$

where $L_P$ is a $P$-norm gradient Lipschitz constant. If we can demonstrate gradient dominance under the dual $P$-norm,

$$\|\nabla f(X)\|_{P*}^2 \geq \mu_P(f(X) - f^\star) \quad \text{where } \mu_P > 0 \text{ and } f^\star = \min f(X), \tag{10}$$

then we have the desired linear convergence

$$f(X - \alpha D) - f^\star \leq [1 - \mu_P\alpha(1 - \alpha L_P/2)] \cdot (f(X) - f^\star) \tag{11}$$

$$= [1 - 1/(2\kappa_P)] \cdot (f(X) - f^\star) \text{ with } \alpha = 1/L_P, \tag{12}$$

in which the condition number $\kappa_P = L_P/\mu_P$ should be upper-bounded. To make the most progress per iteration, we want to pick a metric $\mathbf{P}$ to make the condition number $\kappa_P$ as small as possible.

The best choice of $\mathbf{P}$ for the fastest convergence is simply the Hessian $\nabla^2 f(X)$ itself, but this simply recovers Newton's method, which would forces us to invert a large $nr \times nr$ matrix to compute the search direction $D$ at every iteration. Instead, we look for a *preconditioner* $\mathbf{P}$ that is cheap to apply while still assuring a relatively small condition number $\kappa_P$. The following choice is particularly interesting (the Kronecker product $\otimes$ is defined to satisfy $\text{vec}(AXB^T) = (B \otimes A)\text{vec}(X)$)

$$\mathbf{P} = (X^TX + \eta I_r) \otimes I_n = X^TX \otimes I_n + \eta I_{nr},$$

because the resulting $D = \nabla f(X)(X^T X + \eta I)$ allow us to *interpolate* between regular GD and the ScaledGD of Tong et al. [43]. Indeed, we recover regular GD in the limit $\eta \to \infty$, but as we saw in Section 4, gradient dominance (10) fails to hold, so the condition number $\kappa_P = L_P/\mu_P$ grows unbounded as $\mu_P \to 0$. Instead, setting $\eta = 0$ recovers ScaledGD. The key insight of Tong et al. [43] is that under this choice of $\mathbf{P}$, gradient dominance (10) is guaranteed to hold, with a large value of $\mu_P$ that is independent of the current iterate and the ground truth. But as we will now show, this change of metric can magnify the Lipschitz constant $L_P$ by a factor of $\lambda_{\min}^{-1}(X^T X)$, so the condition number $\kappa_P = L_P/\mu_P$ becomes unbounded in the over-parameterized regime.

**Lemma 2** (Lipschitz-like inequality)**.** *Let* $\|D\|_P = \|D(X^T X + \eta)^{1/2}\|_F$. *Then we have*

$$f(X + D) \le f(X) + \langle \nabla f(X), D \rangle + \frac{1}{2} L_P(X, D) \|D\|_P^2$$

*where*

$$L_P(X, D) = 2(1 + \delta) \left[ 4 + \frac{2\|XX^T - M^\star\|_F + 4\|D\|_P}{\lambda_{\min}(X^T X) + \eta} + \left( \frac{\|D\|_P}{\lambda_{\min}(X^T X) + \eta} \right)^2 \right]$$

**Lemma 3** (Bounded gradient)**.** *For the search direction* $D = \nabla f(X)(X^T X + \eta I)^{-1}$, *we have* $\|D\|_P^2 = \|\nabla f(X)\|_{P*}^2 \le 16(1 + \delta) f(X)$.

The proofs of Lemma 2 and Lemma 3 follows from straightforward linear algebra, and can be found in the Appendix. Substituting Lemma 3 into Lemma 2, we see for ScaledGD (with $\eta = 0$) that the Lipschitz-like constant is bounded as follows

$$L_P(X, D) \lesssim \left( \|XX^T - M^\star\|_F / \lambda_{\min}(X^T X) \right)^2. \tag{13}$$

In the exact rank case $r = r^\star$, the distance of $X$ from singularity can be lower-bounded, within a "good" neighborhood of the ground truth, since $\lambda_{\min}(X^T X) = \lambda_r(X^T X)$ and

$$\|XX^T - M^\star\|_F \le \rho \lambda_r(M^\star), \quad \rho < 1 \implies \lambda_r(X^T X) \ge (1 - \rho)\lambda_r(M^\star) > 0. \tag{14}$$

Within this "good" neighborhood, substituting (14) into (13) yields a Lipschitz constant $L_P$ that depends only on the radius $\rho$. The resulting iterations converge rapidly, independent of any ill-conditioning in the model $XX^T$ nor in the ground-truth $M^\star$. In turn, ScaledGD can be initialized within the the good neighborhood using spectral initialization (see Proposition 6 below).

In the over-parameterized case $r > r^\star$, however, the iterate $X$ must become singular in order for $XX^T$ to converge to $M^\star$, and the radius of the "good" neighorhood reduces to zero. The ScaledGD direction guarantees a large linear progress no matter how singular $X$ may be, but the method may not be able to take a substantial step in this direction if $X$ becomes singular too quickly. To illustrate: the algorithm would fail entirely if it lands at on a point where $\lambda_{\min}(X^T X) = 0$ but $XX^T \neq M^\star$.

While regular GD struggles to make the smallest eigenvalues of $XX^T$ converge to zero, ScaledGD gets in trouble by making these eigenvalues converge quickly. In finding a good mix between these two methods, an intuitive idea is to use the damping parameter $\eta$ to control the rate at which $X$ becomes singular. More rigorously, we can pick an $\eta \approx \|XX^T - ZZ^T\|_F$ and use Lemma 2 to keep the Lipschitz constant $L_P$ bounded. Substituting Lemma 3 into Lemma 2 and using RIP to upper-bound $f(X) \le (1 + \delta)\|XX^T - M^\star\|_F^2$ and $\delta \le 1$ yields

$$\eta \ge C_{\text{lb}} \|XX^T - ZZ^T\|_F \implies L_P(X, D) \le 16 + 136/C_{\text{lb}} + 256/C_{\text{lb}}^2. \tag{15}$$

However, the gradient dominance condition (10) will necessarily fail if $\eta$ is set too large. Our main result in this paper is that keeping $\eta$ within the same order of magnitude as the error norm $\|XX^T - ZZ^T\|_F$ is enough to maintain gradient dominance. The following is the noiseless version of this result.

**Theorem 4** (Noiseless gradient dominance)**.** *Let* $\min_X f(X) = 0$ *for* $M^\star \neq 0$. *Suppose that* $X$ *satisfies* $f(X) \le \rho^2 \cdot (1 - \delta)\lambda_{r^\star}^2(M^\star)$ *with radius* $\rho > 0$ *that satisfies* $\rho^2/(1 - \rho^2) \le (1 - \delta^2)/2$. *Then, we have*

$$\eta \le C_{\text{ub}} \|XX^T - ZZ^T\|_F \quad \implies \quad \|\nabla f(X)\|_{P*}^2 \ge 2\mu_P f(X)$$

*where*

$$\mu_P = \left( \sqrt{\frac{1 + \delta^2}{2}} - \delta \right)^2 \cdot \min \left\{ \left( \frac{C_{\text{ub}}}{\sqrt{2} - 1} \right)^{-1}, \left( 1 + 2C_{\text{ub}} \sqrt{\frac{(r - r^\star)}{1 - \delta^2}} \right)^{-1} \right\} \tag{16}$$

The proof of Theorem 4 is difficult. It requires setting up an optimization over the gradient $P$-norms, and then optimizing over all RIP operators $\mathcal{A}$ using a recent technique of Zhang et al. [50]; we defer the details to the Appendix. In the noiseless case, we get a good estimate of $\eta$ for free as a consequence of RIP:

$$\eta = \sqrt{f(X)} \implies \sqrt{1-\delta}\|XX^T - M^\star\|_F \le \eta \le \sqrt{1+\delta}\|XX^T - M^\star\|_F.$$

Repeating (8)-(12) with Lemma 2, (15) and (16) yields our main result below.

**Corollary 5** (Linear convergence)**.** *Let $X$ satisfy the same initial conditions as in Lemma 4. The search direction $D = \nabla f(X)(X^TX + \eta I)^{-1}$ with damping parameter $\eta = \sqrt{f(X)}$ and step-size $\alpha \le 1/L_P$ yields*

$$f(X - \alpha D) \le (1 - \alpha\mu_P/2)\, f(X)$$

*where $L_P$ is as in (15) with $C_{\text{lb}} = \sqrt{1-\delta}$ and $\mu_P$ is as in (16) with $C_{\text{ub}} = \sqrt{1+\delta}$.*

For a fixed RIP constant $\delta$, Theorem 5 says that PrecGD converges at a linear rate that is independent of the current iterate $X$, and also independent of possible ill-conditioning in the ground truth. However, it does require an initial point $X_0$ that satisfies

$$\|\mathcal{A}(X_0X_0^T - M^*)\|^2 < \rho^2(1-\delta)\lambda_{r^*}(M^\star)^2 \tag{17}$$

with a radius $\rho > 0$ satisfying $\rho^2/(1-\rho^2) \le (1-\delta^2)/2$. Such an initial point can be found using spectral initialization, even if the measurements are tainted with noise. Concretely, we choose the initial point $X_0$ as

$$X_0 = \mathcal{P}_r\left(\frac{1}{m}\sum_{i=1}^m y_iA_i\right) \text{ where } \mathcal{P}_r(M) = \arg\min_{X\in\mathbb{R}^{n\times r}} \|XX^T - M\|_F, \tag{18}$$

where we recall that $y = \mathcal{A}(M^\star) + \epsilon$ are the $m$ possibly noisy measurements collected of the ground truth, and that the rank-$r$ projection operator can be efficiently implemented with a singular value decomposition. The proof of the following proposition can be found in the appendix.

**Proposition 6** (Spectral Initialization)**.** *Suppose that $\delta \le (8\kappa\sqrt{r^*})^{-1}$ and $m \gtrsim \frac{1+\delta}{1-\delta}\frac{\sigma^2 rn\log n}{\rho^2\lambda_{r^\star}^2(M^\star)}$ where $\kappa = \lambda_1(M^\star)/\lambda_{r^*}(M^\star)$. Then, with high probability, the initial point $X_0$ produced by (18) satisfies the radius condition (17).*

However, if the measurements $y$ are noisy, then $\sqrt{f(X)} = \|\mathcal{A}(XX^T - M^\star) + \varepsilon\|$ now gives a biased estimate of our desired damping parameter $\eta$. In the next section, we show that a good choice of $\eta_k$ is available based on an approximation of the noise variance.

## 6 Extension to Noisy Setting

In this section, we extend our analysis to the matrix sensing with noisy measurements. Our main goal is to show that, with a proper choice of the damping coefficient $\eta$, the proposed algorithm converges linearly to an "optimal" estimation error.

**Theorem 7** (Noisy measurements with optimal $\eta$)**.** *Suppose that the noise vector $\epsilon \in \mathbb{R}^m$ has sub-Gaussian entries with zero mean and variance $\sigma^2 = \frac{1}{m}\sum_{i=1}^m \mathbb{E}[\epsilon_i^2]$. Moreover, suppose that $\eta_k = \frac{1}{\sqrt{m}}\|\mathcal{A}(X_kX_k^T - M^*)\|$, for $k = 0, 1, \ldots, K$, and that the initial point $X_0$ satisfies $\|\mathcal{A}(X_0X_0^T - M^*)\|^2 < \rho^2(1-\delta)\lambda_{r^*}(M^\star)^2$. Consider $k^* = \arg\min_k \eta_k$, and suppose that the step-size $\alpha \le 1/L$, where $L > 0$ is a constant that only depends on $\delta$. Then, with high probability, we have*

$$\|X_{k^*}X_{k^*}^T - M^\star\|_F^2 \lesssim \max\left\{\frac{1+\delta}{1-\delta}\left(1 - \alpha\frac{\mu_P}{2}\right)^K \|X_0X_0^T - M^*\|_F^2, \mathcal{E}_{stat}\right\}, \tag{19}$$

*where $\mathcal{E}_{stat} := \frac{\sigma^2 nr\log n}{\mu_P(1-\delta)m}$.*

Assuming fixed parameters for the problem, the above theorem shows that PrecGD outputs a solution with an estimation error of $\mathcal{O}(\mathcal{E}_{stat})$ in $\mathcal{O}(\log(1/\mathcal{E}_{stat}))$ iterations. Moreover, the error $\mathcal{O}(\mathcal{E}_{stat})$ is minimax optimal (modulo logarithmic factors), and cannot be improved significantly. In particular, Candes and Plan [6] showed that *any* other estimator $\widehat{X}$ must satisfy $\|\widehat{X}\widehat{X}^T - M^*\|_F^2 \gtrsim \sigma^2 nr/m$

with non-negligible probability. The classical methods for achieving this minimax rate suffer from computationally-prohibitive per iteration costs [14, 37, 33]. Regular gradient descent alleviates this issue at the expense of a slower convergence rate of $\mathcal{O}(\sqrt{1/\mathcal{E}_{stat}})$ [54]. Our proposed PrecGD achieves the best of both worlds: it converges to the minimax optimal error with cheap per-iteration complexity of $\mathcal{O}(nr^2 + r^3)$, while benefiting from an exponentially faster convergence rate than regular gradient descent in the over-parameterized regime.

Theorem 7 highlights the critical role of the damping coefficient $\eta$ in the guaranteed linear convergence of the algorithm. In the noiseless regime, we showed in the previous section that an "optimal" choice $\eta = \sqrt{f(X)}$ is available for free. In the noisy setting, however, the same choice of $\eta$ becomes biased by the noise variance, and is therefore no longer optimal. As is typically the case for regularized estimation methods [16, 11, 22], selecting the ideal parameter would amount to some kind of *resampling*, such as via cross-validation or bootstrapping [21, 17, 15], which is generally expensive to implement and use in practice. As an alternative approach, we show in our next theorem that a good choice of $\eta$ is available based on an approximation of the noise variance $\sigma^2$.

**Theorem 8** (Noisy measurements with variance proxy)**.** *Suppose that the noise vector $\epsilon \in \mathbb{R}^m$ has sub-Gaussian entries with zero mean and variance $\sigma^2 = \frac{1}{m}\sum_{i=1}^m \mathbb{E}[\epsilon_i^2]$. Moreover, suppose that $\eta_k = \sqrt{|f(X_k) - \hat{\sigma}^2|}$ for $k = 0, 1, \ldots, K$, where $\hat{\sigma}^2$ is an approximation of $\sigma^2$, and that the initial point $X_0$ satisfies $\|\mathcal{A}(X_0X_0^T - M^*)\|_F^2 < \rho^2(1-\delta)\lambda_{r^*}(M^\star)^2$. Consider $k^* = \arg\min_k \eta_k$, and suppose that the step-size $\alpha \leq 1/L$, where $L > 0$ is a constant that only depends on $\delta$. Then, with high probability, we have*

$$\|X_{k^*}X_{k^*}^T - M^*\|_F^2 \lesssim \max\left\{\frac{1+\delta}{1-\delta}\left(1 - \alpha\frac{\mu_P}{2}\right)^K \|X_0X_0^T - M^*\|_F^2, \mathcal{E}_{stat}, \mathcal{E}_{dev}, \mathcal{E}_{var}\right\}, \quad (20)$$

*where*

$$\mathcal{E}_{stat} := \frac{\sigma^2 nr\log n}{\mu_P(1-\delta)m}, \quad \mathcal{E}_{dev} := \frac{1}{1-\delta}\sqrt{\frac{\sigma^2\log n}{m}}, \quad \mathcal{E}_{var} := (\sigma^2 - \hat{\sigma}^2)^2. \quad (21)$$

In the above theorem, $\mathcal{E}_{dev}$ captures the deviation of the empirical variance $\frac{1}{m}\sum_{i=1}^m \epsilon_i^2$ from its expectation $\sigma^2$. On the other hand, $\mathcal{E}_{var}$ captures the approximation error of the true variance. According to Theorem 8, it is possible to chose the damping factor $\eta_k$ merely based on $f(X)$ and an approximation of $\sigma^2$, at the expense of a suboptimal estimation error rate. In particular, suppose that the noise variance is known precisely, i.e., $\hat{\sigma}^2 = \sigma^2$. Then, the above theorem implies that the estimation error is reduced to

$$\|X_{k^*}X_{k^*}^T - M^*\|_F^2 \lesssim \max\{\mathcal{E}_{stat}, \mathcal{E}_{dev}\} \quad \text{after} \quad \mathcal{O}\left(\log\left(\frac{1}{\max\{\mathcal{E}_{stat}, \mathcal{E}_{dev}\}}\right)\right) \text{ iterations.}$$

If $m$ is not too large, i.e., $m \lesssim \sigma^2 n^2 r^2 \log n$, the estimation error can be improved to $\|X_{k^*}X_{k^*}^T - M^*\|_F^2 \lesssim \mathcal{E}_{stat}$, which is again optimal (modulo logarithmic factors). The suboptimal rate is due to the heavy tail phenomenon arising from the concentration of the noise variance. In particular, one can write

$$f(X) - \sigma^2 = \frac{1}{m}\|\mathcal{A}(XX^T - M^\star)\|^2 + \underbrace{\frac{1}{m}\|\epsilon\|^2 - \sigma^2}_{\text{variance deviation}} + \underbrace{\frac{2}{m}\langle\mathcal{A}(ZZ^T - XX^T), \epsilon\rangle}_{\text{cross-term}} \quad (22)$$

Evidently, $f(X) - \sigma^2$ is in the order of $\frac{1}{m}\|\mathcal{A}(XX^T - M^\star)\|^2$ if both variance deviation and cross-term are dominated by $\frac{1}{m}\|\mathcal{A}(XX^T - M^\star)\|^2$. In the proof of Theorem 8, we show that, with high probability, the variance deviation is upper bounded by $(1-\delta)\mathcal{E}_{dev}$ and it dominates the cross-term. This implies that the choice of $\eta = \sqrt{|f(X) - \sigma^2|}$ behaves similar to $\frac{1}{\sqrt{m}}\|\mathcal{A}(XX^T - M^\star)\|$, and hence, the result of Theorem 7 can be invoked, so long as

$$\frac{1}{m}\|\mathcal{A}(XX^T - M^\star)\|^2 \geq (1-\delta)\|XX^T - M^\star\|_F^2 \gtrsim (1-\delta)\mathcal{E}_{dev}.$$

## 7 Numerical Experiments

Finally, we numerically compare PrecGD on other matrix factorization problems that fall outside of the matrix sensing framework. We consider the $\ell_p$ empirical loss $f_p(X) = \sum_{i=1}^m |\langle A_i, XX^T - M^\star\rangle|^p$

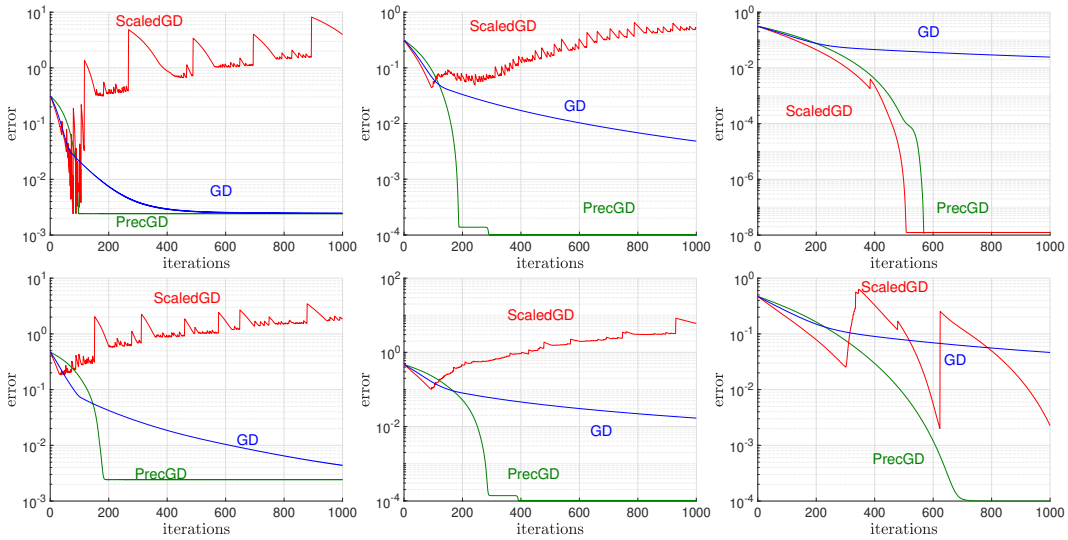


Figure 2: **Nonconvex matrix factorization with the $\ell_p$ empirical loss**. We compare $\ell_p$ matrix sensing with $n = 10$ and $r^\star = 2$ and $\mathcal{A}$ taken from [50]. The ground truth is chosen to be ill-conditioned ($\kappa = 10^4$). All three methods use the same initial point and the same learning rate $\alpha = 10^{-3}$. From left to right, the values for $p$ are $1.1, 1.4$ and $1.7$ respectively. **(Top $r = r^*$)** For all three values of $p$, GD stagnates due to the ill-conditioning of the ground truth. When $p$ is small, ScaledGD suffers from erratic behavior. In contrast, ScaledGD converges superlinearly in all three cases. **(Bottom $r > r^*$)** With $r = 3$, GD again converges slowly and ScaledGD is sporadic due to near-singularity caused by over-parameterization. Once again we see PrecGD converge at a superlinear rate.

for $1 \leq p < 2$, in order to gauge the effectiveness of PrecGD for increasing nonsmooth loss functions. Here, we set the damping parameter $\eta_k = [f_p(X_k)]^{1/p}$ as a heuristic for the error $\|XX^T - M^\star\|_F$. The data matrices $A_1, \ldots, A_m$ were taken from [50, Example 12], the ground truth $M^\star = ZZ^T$ was constructed by sampling each column of $Z \in \mathbb{R}^{n \times r^\star}$ from the standard Gaussian, and then rescaling the last column to achieve a desired condition number. Each method is initialized using a random initial point. Figure 2 compares GD, ScaledGD and PrecGD in the exact and over-parameterized regimes for the $\ell_p$ norm, with $p = 1.1, 1.4$ and $1.7$. By varying the value of $p$, the loss function is made increasing nonsmooth. When $r = r^*$, GD reliably converges, but at a slow rate due to ill-conditioning of the ground truth. ScaledGD exhibits sporadic behavior once $p$ is close to 1, as the loss becomes highly nonsmooth. PrecGD reliably converges; it is able to overcome the ill-conditioning of the ground truth, and also the non-smoothness caused by $p$ being close to 1. When $r > r^*$, GD converges slowly, ScaledGD again becomes sporadic, while PrecGD continues to converge reliably.

## 8 Conclusions

In this paper, we propose a *preconditioned* gradient descent or PrecGD for nonconvex matrix factorization with a comparable per-iteration cost to classical gradient descent. For over-parameterized matrix sensing, gradient descent slows down to a sublinear convergence rate, but PrecGD restores the convergence rate back to linear, while also making the iterations immune to ill-conditioning in the ground truth. Our numerical results find that PrecGD continues to work well for other nonconvex matrix factorization problems outside of classical matrix sensing, and that it exhibits global behavior without actually requiring a good initial point. It remains future work to provide a rigorous justification for these empirical observations.

# References


[1] A. Ahmed, B. Recht, and J. Romberg. Blind deconvolution using convex programming. *IEEE Transactions on Information Theory*, 60(3):1711–1732, 2013.

[2] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM journal on Optimization*, 5(1):13–51, 1995.

[3] S. Bhojanapalli, A. Kyrillidis, and S. Sanghavi. Dropping convexity for faster semi-definite optimization. In *Conference on Learning Theory*, pages 530–582. PMLR, 2016.

[4] S. Bhojanapalli, B. Neyshabur, and N. Srebro. Global optimality of local search for low rank matrix recovery. *arXiv preprint arXiv:1605.07221*, 2016.

[5] J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on optimization*, 20(4):1956–1982, 2010.

[6] E. J. Candes and Y. Plan. Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Transactions on Information Theory*, 57(4): 2342–2359, 2011.

[7] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

[8] E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2010.

[9] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):1–37, 2011.

[10] E. J. Candes, X. Li, and M. Soltanolkotabi. Phase retrieval via wirtinger flow: Theory and algorithms. *IEEE Transactions on Information Theory*, 61(4):1985–2007, 2015.

[11] G. C. Cawley. Leave-one-out cross-validation based model selection criteria for weighted ls-svms. In *The 2006 IEEE international joint conference on neural network proceedings*, pages 1661–1668. IEEE, 2006.

[12] V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011.

[13] J. Chen and X. Li. Memory-efficient kernel pca via partial matrix sampling and nonconvex optimization: a model-free analysis of local minima. *arXiv preprint arXiv:1711.01742*, 2017.

[14] Y. Chen and M. J. Wainwright. Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees. *arXiv preprint arXiv:1509.03025*, 2015.

[15] D. R. Cox and D. V. Hinkley. *Theoretical statistics*. CRC Press, 1979.

[16] E. De Vito, A. Caponnetto, and L. Rosasco. Model selection for regularized least-squares algorithm in learning theory. *Foundations of Computational Mathematics*, 5(1):59–85, 2005.

[17] B. Efron and R. J. Tibshirani. *An introduction to the bootstrap*. CRC press, 1994.

[18] R. Ge, F. Huang, C. Jin, and Y. Yuan. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on learning theory*, pages 797–842. PMLR, 2015.

[19] R. Ge, J. D. Lee, and T. Ma. Matrix completion has no spurious local minimum. *arXiv preprint arXiv:1605.07272*, 2016.

[20] R. Ge, C. Jin, and Y. Zheng. No spurious local minima in nonconvex low rank problems: A unified geometric analysis. In *International Conference on Machine Learning*, pages 1233–1242. PMLR, 2017.

[21] P. I. Good. *Resampling methods*. Springer, 2006.

[22] J. Guo, E. Levina, G. Michailidis, and J. Zhu. Joint estimation of multiple graphical models. *Biometrika*, 98(1):1–15, 2011.

[23] M. Hardt and M. Wootters. Fast matrix completion without the condition number. In *Conference on learning theory*, pages 638–678. PMLR, 2014.

[24] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 665–674, 2013.

[25] C. Jin, R. Ge, P. Netrapalli, S. M. Kakade, and M. I. Jordan. How to escape saddle points efficiently. In *International Conference on Machine Learning*, pages 1724–1732. PMLR, 2017.

[26] R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from a few entries. *IEEE transactions on information theory*, 56(6):2980–2998, 2010.

[27] Q. Li, Z. Zhu, and G. Tang. The non-convex geometry of low-rank matrix optimization. *Information and Inference: A Journal of the IMA*, 8(1):51–96, 2019.

[28] Y. Li, T. Ma, and H. Zhang. Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. In *Conference On Learning Theory*, pages 2–47. PMLR, 2018.

[29] S. Ling and T. Strohmer. Self-calibration and biconvex compressive sensing. *Inverse Problems*, 31(11):115002, 2015.

[30] X. Luo, M. Zhou, Y. Xia, and Q. Zhu. An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. *IEEE Transactions on Industrial Informatics*, 10(2):1273–1284, 2014.

[31] J. Ma and S. Fattahi. Implicit regularization of sub-gradient method in robust matrix recovery: Don't be afraid of outliers. *arXiv preprint arXiv:2102.02969*, 2021.

[32] R. Meka, P. Jain, and I. S. Dhillon. Guaranteed rank minimization via singular value projection. *arXiv preprint arXiv:0909.5457*, 2009.

[33] S. Negahban and M. J. Wainwright. Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *The Annals of Statistics*, pages 1069–1097, 2011.

[34] P. Netrapalli, U. Niranjan, S. Sanghavi, A. Anandkumar, and P. Jain. Non-convex robust pca. *arXiv preprint arXiv:1410.7660*, 2014.

[35] J. Nocedal and S. Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[36] B. O'donoghue, E. Chu, N. Parikh, and S. Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. *Journal of Optimization Theory and Applications*, 169(3): 1042–1068, 2016.

[37] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[38] A. Singer. Angular synchronization by eigenvectors and semidefinite programming. *Applied and computational harmonic analysis*, 30(1):20–36, 2011.

[39] M. Soltanolkotabi. Structured signal recovery from quadratic measurements: Breaking sample complexity barriers via nonconvex optimization. *IEEE Transactions on Information Theory*, 65 (4):2374–2400, 2019.

[40] J. Sun, Q. Qu, and J. Wright. Complete dictionary recovery over the sphere i: Overview and the geometric picture. *IEEE Transactions on Information Theory*, 63(2):853–884, 2016.

[41] J. Sun, Q. Qu, and J. Wright. A geometric analysis of phase retrieval. *Foundations of Computational Mathematics*, 18(5):1131–1198, 2018.

[42] R. Sun and Z.-Q. Luo. Guaranteed matrix completion via non-convex factorization. *IEEE Transactions on Information Theory*, 62(11):6535–6579, 2016.

[43] T. Tong, C. Ma, and Y. Chi. Accelerating ill-conditioned low-rank matrix estimation via scaled gradient descent. *arXiv preprint arXiv:2005.08898*, 2020.

[44] S. Tu, R. Boczar, M. Simchowitz, M. Soltanolkotabi, and B. Recht. Low-rank solutions of linear matrix equations via procrustes flow. In *International Conference on Machine Learning*, pages 964–973. PMLR, 2016.

[45] Z. Wen, D. Goldfarb, and W. Yin. Alternating direction augmented lagrangian methods for semidefinite programming. *Mathematical Programming Computation*, 2(3-4):203–230, 2010.

[46] X. Yi, D. Park, Y. Chen, and C. Caramanis. Fast algorithms for robust pca via gradient descent. *arXiv preprint arXiv:1605.07784*, 2016.

[47] K. Yu, S. Zhu, J. Lafferty, and Y. Gong. Fast nonparametric matrix factorization for large-scale collaborative filtering. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 211–218, 2009.

[48] R. Zhang, C. Josz, S. Sojoudi, and J. Lavaei. How much restricted isometry is needed in nonconvex matrix recovery? 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. URL https://proceedings.neurips.cc/paper/2018/file/f8da71e562ff44a2bc7edf3578c593da-Paper.pdf.

[49] R. Y. Zhang. Sharp global guarantees for nonconvex low-rank matrix recovery in the overparameterized regime. *arXiv preprint arXiv:2104.10790*, 2021.

[50] R. Y. Zhang, S. Sojoudi, and J. Lavaei. Sharp restricted isometry bounds for the inexistence of spurious local minima in nonconvex matrix recovery. *Journal of Machine Learning Research*, 20(114):1–34, 2019.

[51] Q. Zheng and J. Lafferty. A convergent gradient descent algorithm for rank minimization and semidefinite programming from random linear measurements. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper/2015/file/32bb90e8976aab5298d5da10fe66f21d-Paper.pdf.

[52] Q. Zheng and J. Lafferty. A convergent gradient descent algorithm for rank minimization and semidefinite programming from random linear measurements. *arXiv preprint arXiv:1506.06081*, 2015.

[53] Y. Zheng, G. Fantuzzi, A. Papachristodoulou, P. Goulart, and A. Wynn. Chordal decomposition in operator-splitting methods for sparse semidefinite programs. *Mathematical Programming*, 180(1):489–532, 2020.

[54] J. Zhuo, J. Kwon, N. Ho, and C. Caramanis. On the computational and statistical complexity of over-parameterized matrix sensing. *arXiv preprint arXiv:2102.02756*, 2021.

## 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]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (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]
   (b) Did you include complete proofs of all theoretical results? [Yes]
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)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (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)? [N/A]
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]