# Global Convergence of Non-Convex Gradient Descent for Computing Matrix Squareroot

**Prateek Jain**
Microsoft Research India

**Chi Jin**
UC Berkeley

**Sham M. Kakade**
University of Washington

**Praneeth Netrapalli**
Microsoft Research India

## Abstract

While there has been a significant amount of work studying gradient descent techniques for non-convex optimization problems over the last few years, all existing results establish either *local convergence with good rates* or *global convergence with highly suboptimal rates*, for many problems of interest. In this paper, we take the first step in getting the best of both worlds – establishing global convergence and obtaining a good rate of convergence for the problem of computing squareroot of a positive definite (PD) matrix, which is a widely studied problem in numerical linear algebra with applications in machine learning and statistics among others.

Given a PD matrix $\mathbf{M}$ and a PD starting point $\mathbf{U_0}$, we show that gradient descent with appropriately chosen stepsize finds an $\epsilon$-accurate squareroot of $\mathbf{M}$ in $\mathcal{O}\left(\alpha \log(\left\|\mathbf{M}-\mathbf{U_0}^2\right\|_F/\epsilon)\right)$ iterations, where $\alpha \triangleq (\max\{\|\mathbf{U_0}\|_2^2, \|\mathbf{M}\|_2\} / \min\{\sigma_{\min}^2(\mathbf{U_0}), \sigma_{\min}(\mathbf{M})\})^{3/2}$. Our result is the first to establish global convergence for this problem and that it is robust to errors in each iteration.

A key contribution of our work is the general proof technique which we believe should further excite research in understanding deterministic and stochastic variants of simple non-convex gradient descent algorithms with *good global convergence rates* for other problems in machine learning and numerical linear algebra.

Appearing in Proceedings of the $20^{th}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2017, Fort Lauderdale, Florida, USA. JMLR: W&CP volume 54. 

## 1 Introduction

Given that a large number of problems and frameworks in machine learning are non-convex optimization problems (examples include non-negative matrix factorization [Lee and Seung, 2001], sparse coding [Aharon et al., 2006], matrix sensing [Recht et al., 2010], matrix completion [Koren et al., 2009], phase retrieval [Netrapalli et al., 2015] etc.), in the last few years, there has been an increased interest in designing efficient non-convex optimization algorithms. Several recent works establish *local convergence* to the global optimum for problems such as matrix sensing [Jain et al., 2013, Tu et al., 2015], matrix completion [Jain and Netrapalli, 2014, Sun and Luo, 2015], phase retrieval [Candes et al., 2015], sparse coding [Agarwal et al., 2013] and so on (and hence, require careful initialization). However, despite strong empirical evidence, none of these results have been able to establish *global convergence*. On the other hand some other recent works [Nesterov and Polyak, 2006, Ge et al., 2015, Lee et al., 2016, Sun et al., 2015] establish the global convergence of gradient descent methods to local minima for a large class of non-convex problems but the results they obtain are quite suboptimal compared to the local convergence results mentioned above. In other words, results that have very good rates are only local (and results that are global do not have very good rates).

Therefore, a natural and important question is if gradient descent actually has a *good global convergence rate* when applied to specific and important functions that are of interest in machine learning. Apart from theoretical implications, such a result is also important in practice since a) finding a good initialization might be difficult and b) local convergence results are inherently difficult to extend to stochastic algorithms due to noise.

In this work, we answer the above question in affirmative for the problem of computing square root of a positive definite (PD) matrix $\mathbf{M}$: i.e., $\min_{\mathbf{U}\succeq 0} f(\mathbf{U})$ where $f(\mathbf{U}) = \|\mathbf{M}-\mathbf{U}^2\|_F^2$. This problem in itself is a fundamental one and arises in several contexts such

as computation of the matrix sign function [Higham, 2008], computation of data whitening matrices, signal processing applications [Kaminski et al., 1971, Carlson, 1990, Van Der Merwe and Wan, 2001, Tippett et al., 2003] and so on.

### 1.1 Related work

Given the importance of computing the matrix squareroot, there has been a tremendous amount of work in the numerical linear algebra community focused on this problem [Björck and Hammarling, 1983, Higham, 1986, 1987, 1997, Meini, 2004]. For a detailed list of references, see Chapter 6 in Higham's book [Higham, 2008].

The basic component of most these algorithms is the Newton's method to find the square root of a positive number. Given a positive number $m$ and a positive starting point $u_0$, Newton's method gives rise to the iteration

$$u_{t+1} \leftarrow \frac{1}{2}\left(u_t + \frac{m}{u_t}\right). \tag{1}$$

It can be shown that the iterates converge to $\sqrt{m}$ at a quadratic rate (i.e., $\epsilon$-accuracy in $\log\log\frac{1}{\epsilon}$ iterations). The extension of this approach to the matrix case is not straight forward due to non commutativity of matrix multiplication. For instance, if $\mathbf{M}$ and $\mathbf{U_t}$ were matrices, it is not clear if $\frac{m}{u_t}$ should be replaced by $\mathbf{U_t}^{-1}\mathbf{M}$ or $\mathbf{M}\mathbf{U_t}^{-1}$ or something else. One approach to overcome this issue is to select $\mathbf{U_0}$ carefully to ensure commutativity through all iterations [Higham, 1986, 1997, Meini, 2004], for example, $\mathbf{U_0} = \mathbf{M}$ or $\mathbf{U_0} = \mathbf{I}$. However, commutativity is a brittle property and small numerical errors in an iteration itself can result in loss of commutativity. Although a lot of work since, has focused on designing stable iterations that are inspired by Eq.(1) [Higham, 1986, 1997, Meini, 2004], and has succeeded in making it robust in practice, no provable robustness guarantees are known in the presence of repeated errors. Similarly, another recent approach by Sra [2015] uses geometric optimization to solve the matrix squareroot problem but their analysis also does not address the stability or robustness to numerical or statistical errors (if we see a noisy version of $\mathbf{M}$) .

Another approach to solve the matrix square-root problem is to use the eigenvalue decomposition (EVD) and then take square-root of the eigenvalues. To the best of our knowledge, state-of-the-art computation complexity for computing the EVD of a matrix (in the real arithmetic model of computation) is due to Pan et al. [1998], which is $\mathcal{O}\left(n^\omega \log n + n\log^2 n \log\log\frac{1}{\epsilon}\right)$ for matrices with distinct eigenvalues. Though the result is close to optimal (in reducing the EVD to matrix multiplication), the algorithm and the analysis are quite complicated. For instance robustness of these methods to errors is not well understood. As mentioned above however, our focus is to understand if local search techniques like gradient descent (which are often applied to several non-convex optimization procedures) indeed avoid saddle points and local minima, and can guide the solution to global optimum.

Finally, as we mentioned earlier, Ge et al. [2015], Lee et al. [2016] give some recent results on global convergence for general non-convex problems which can be applied to matrix squareroot problem. While Lee et al. [2016] prove only asymptotic behavior of gradient descent without any rate, applying the result of Ge et al. [2015] gives us a runtime of $\mathcal{O}\left(n^{10}/\mathrm{poly}\left(\epsilon\right)\right)$[1], which is highly suboptimal in terms of its dependence both on $n$ and on $\epsilon$.

### 1.2 Our contribution

In this paper, we propose doing gradient descent on the following non-convex formulation:

$$\min_{\mathbf{U}\in\mathbb{R}^{n\times n};\mathbf{U}\succeq 0} \left\|\mathbf{M}-\mathbf{U}^2\right\|_F^2. \tag{2}$$

We show that if the starting point $\mathbf{U_0}$ is chosen to be a positive definite matrix, our algorithm converges to the global optimum of Eq.(2) at a geometric rate. In order to state our runtime, we make the following notation:

$$\alpha \triangleq \left(\frac{\max\left(\left\|\mathbf{U_0}\right\|_2, \sqrt{\left\|\mathbf{M}\right\|_2}\right)}{\min\left(\sigma_{\min}\left(\mathbf{U_0}\right), \sqrt{\sigma_{\min}\left(\mathbf{M}\right)}\right)}\right)^3, \tag{3}$$

where $\sigma_{\min}\left(\mathbf{U_0}\right)$ and $\left\|\mathbf{U_0}\right\|_2$ are the minimum singular value and operator norm respectively of the starting point $\mathbf{U_0}$, and $\sigma_{\min}\left(\mathbf{M}\right)$ and $\left\|\mathbf{M}\right\|_2$ are those of $\mathbf{M}$. Our result says that gradient descent converges $\epsilon$ close to the optimum of Eq.(2) in $\mathcal{O}\left(\alpha\log\frac{\left\|\mathbf{M}-\mathbf{U_0}^2\right\|_F}{\epsilon}\right)$ iterations. Each iteration involves doing only three matrix multiplications and no inversions or least-squares. So the total runtime of our algorithm is $\mathcal{O}\left(n^\omega\alpha\log\frac{\left\|\mathbf{M}\right\|_F}{\epsilon}\right)$, where $\omega < 2.373$ is the matrix multiplication exponent[Williams, 2012]. As a byproduct of our global convergence guarantee, we obtain the robustness of our algorithm to numerical errors *for free*. In particular, we show that our algorithm is

[1]For optimization problem of dimension $d$, Ge et al. [2015] proves convergence in the number of iteration of $\mathcal{O}\left(d^4\right)$, with $\mathcal{O}\left(d\right)$ computation per iteration. In matrix squareroot problem $d = n^2$, which gives total $\mathcal{O}\left(n^{10}\right)$ dependence.

| **Method** | **Runtime** | Global convergence | Provable robustness |
|---|---|---|---|
| **Gradient descent (this paper)** | $\mathcal{O}\left(\alpha n^{\omega}\log\frac{1}{\epsilon}\right)$ | ✓ | ✓ |
| Stochastic gradient descent [Ge et al., 2015] | $\mathcal{O}\left(n^{10}/\mathrm{poly}\left(\epsilon\right)\right)$ | ✓ | ✓ |
| Newton variants [Higham, 2008] | $\mathcal{O}\left(n^{\omega}\log\log\frac{1}{\epsilon}\right)$ | × | × |
| EVD (algebraic [Pan et al., 1998]) | $\mathcal{O}\left(n^{\omega}\log n + n\log^2 n\log\log\frac{1}{\epsilon}\right)$ | Not iterative | × |
| EVD (power method [Golub and Van Loan, 2012]) | $\mathcal{O}\left(n^{3}\log\frac{1}{\epsilon}\right)$ | Not iterative | × |

Table 1: Comparison of our result to existing ones. Here $\omega$ is the matrix multiplication exponent and $\alpha$ is our convergence rate parameter defined in Eq.(3). We show that our method enjoys global convergence and is also provably robust to arbitrary bounded errors in each iteration. In contrast, Newton variants only have local convergence and their robustness to errors in multiple iterations is not known. Robustness of methods based on eigenvalue decomposition is also not well understood.

robust to errors in multiple steps in the sense that if each step has an error of at most $\delta$, then our algorithm achieves a limiting accuracy of $\mathcal{O}\left(\alpha\sqrt{\|\mathbf{M}\|_2}\delta\right)$. Another nice feature of our algorithm is that it is based purely on matrix multiplications, where as most existing methods require matrix inversion or solving a system of linear equations. An unsatisfactory part of our result however is the dependence on $\alpha \geq \kappa^{3/2}$, where $\kappa$ is the condition number of $\mathbf{M}$. We prove a lower bound of $\Omega\left(\kappa\right)$ iterations for our method which tells us that the dependence on problem parameters in our result is not a weakness in our analysis.

**Outline**: In Section 2, we will briefly set up the notation we will use in this paper. In Section 3, we will present our algorithm, approach and main results. We will present the proof of our main result in Section 4 and conclude in Section 5. The proofs of remaining results can be found in the Appendix.

## 2 Notation

Let us briefly introduce the notation we will use in this paper. We use boldface lower case letters $(\mathbf{v}, \mathbf{w}, \ldots)$ to denote vectors and boldface upper case letters $(\mathbf{M}, \mathbf{X}, \ldots)$ to denote matrices. $\mathbf{M}$ denotes the input matrix we wish to compute the squareroot of. $\sigma_i\left(\mathbf{A}\right)$ denotes the $i^{\text{th}}$ singular value of $\mathbf{A}$. $\sigma_{\min}\left(\mathbf{A}\right)$ denotes the smallest singular value of $\mathbf{A}$. $\kappa\left(\mathbf{A}\right)$ denotes the condition number of $\mathbf{A}$ i.e., $\frac{\|\mathbf{A}\|_2}{\sigma_{\min}(\mathbf{A})}$. $\kappa$ without an argument denotes $\kappa\left(\mathbf{M}\right)$. $\lambda_i\left(\mathbf{A}\right)$ denotes the $i^{\text{th}}$ largest eigenvalue of $\mathbf{A}$ and $\lambda_{\min}(\mathbf{A})$ denotes the smallest eigenvalue of $\mathbf{A}$.

**Algorithm 1** Gradient descent for matrix square root

**Input:** $\mathbf{M}$, PD matrix $\mathbf{U_0}, \eta, T$
**Output:** $\mathbf{U}$
  **for** $t = 0, \cdots, T-1$ **do**
    $\mathbf{U_{t+1}} = \mathbf{U_t} - \eta\left(\mathbf{U_t}^2 - \mathbf{M}\right)\mathbf{U_t} - \eta\mathbf{U_t}\left(\mathbf{U_t}^2 - \mathbf{M}\right)$
  **end for**
  **Return** $\mathbf{U_T}$.

## 3 Our Results

In this section, we present our guarantees and the high-level approach for the analysis of Algorithm 1 which is just gradient descent on the non-convex optimization problem:

$$\min_{\mathbf{U}\in\mathbb{R}^{n\times n};\mathbf{U}\succeq 0}\left\|\mathbf{M}-\mathbf{U}^2\right\|_F^2. \tag{4}$$

We first present a warmup analysis, where we assume that all the iterates of Algorithm 1 commute with $\mathbf{M}$. Later, in Section 3.2 we present our approach to analyze Algorithm 1 for any general starting point $\mathbf{U_0}$. We provide formal guarantees in Section 3.3.

### 3.1 Warmup – Analysis with commutativity

In this section, we will give a short proof of convergence for Algorithm 1, when we ensure that all iterates commute with $\mathbf{M}$.

**Lemma 3.1.** *There exists a constant $c$ such that if $\eta < \frac{c}{\|\mathbf{M}\|_2}$, and $\mathbf{U_0}$ is chosen to be $\sqrt{\|\mathbf{M}\|_2}\cdot\mathbf{I}$, then $\mathbf{U_t}$ in Algorithm 1 satisfies:*

$$\left\|\mathbf{U_t}^2-\mathbf{M}\right\|_F^2 \leq \exp\left(-2\eta\sigma_{\min}\left(\mathbf{M}\right)t\right)\left\|\mathbf{U_0}^2-\mathbf{M}\right\|_F^2.$$

*Proof.* Since $\mathbf{U_0} = \sqrt{\|\mathbf{M}\|_2}\mathbf{I}$ has the same eigenvectors as $\mathbf{M}$, it can be seen by induction that $\mathbf{U_t}$ has the same eigenvectors as $\mathbf{M}$ for every $t$. Every singular value $\sigma_i(\mathbf{U_{t+1}})$ can be written as

$$\sigma_i(\mathbf{U_{t+1}}) = \left(1 - 2\eta\left(\sigma_i(\mathbf{U_t})^2 - \sigma_i(\mathbf{M})\right)\right)\sigma_i(\mathbf{U_t}). \tag{5}$$

Firstly, this tells us that $\|\mathbf{U_t}\|_2 < \sqrt{2\|\mathbf{M}\|_2}$ for every $t$. Verifying this is easy using induction. The statement holds for $t = 0$ by hypothesis. Assuming it holds for $\mathbf{U_t}$, the induction step follows by considering the two cases $\|\mathbf{U_t}\|_2 \leq \sqrt{\|\mathbf{M}\|_2}$ and $\sqrt{\|\mathbf{M}\|_2} < \|\mathbf{U_t}\|_2 < \sqrt{2\|\mathbf{M}\|_2}$ separately and using the assumption that $\eta < \frac{c}{\|\mathbf{M}\|_2}$. A similar induction argument also tells us that $\sigma_i(\mathbf{U_t}) > \sqrt{\frac{\sigma_i(\mathbf{M})}{2}}$. Eq.(5) can now be used to yield the following convergence equation:

$$\begin{aligned}
&\left|\sigma_i(\mathbf{U_{t+1}})^2 - \sigma_i(\mathbf{M})\right| \\
&= \left|\sigma_i(\mathbf{U_t})^2 - \sigma_i(\mathbf{M})\right| \cdot \Big(1 - 4\eta\sigma_i(\mathbf{U_t})^2 \\
&\qquad +4\eta^2\sigma_i(\mathbf{U_t})^2\left(\sigma_i(\mathbf{U_t})^2 - \sigma_i(\mathbf{M})\right)\Big) \\
&\leq \left|\sigma_i(\mathbf{U_t})^2 - \sigma_i(\mathbf{M})\right| \cdot \Big(1 - 4\eta\sigma_i(\mathbf{U_t})^2 \\
&\qquad +8\eta^2\sigma_i(\mathbf{U_t})^2\|\mathbf{M}\|_2\Big) \\
&\leq \left(1 - 2\eta\sigma_i(\mathbf{U_t})^2\right)\left|\sigma_i(\mathbf{U_t})^2 - \sigma_i(\mathbf{M})\right| \\
&\leq \exp\left(-\eta\sigma_{\min}(\mathbf{M})\right)\left|\sigma_i(\mathbf{U_t})^2 - \sigma_i(\mathbf{M})\right|,
\end{aligned}$$

where we used the hypothesis on $\eta$ in the last two steps. Using induction gives us

$$\left|\sigma_i(\mathbf{U_t})^2 - \mathbf{M}\right| \leq \exp\left(-\eta\sigma_{\min}(\mathbf{M})\,t\right)\left|\sigma_i(\mathbf{U_0})^2 - \mathbf{M}\right|.$$

This can now be used to prove the lemma:

$$\begin{aligned}
&\left\|\mathbf{U_t}^2 - \mathbf{M}\right\|_F^2 = \sum_i\left(\sigma_i(\mathbf{U_t})^2 - \sigma_i(\mathbf{M})\right)^2 \\
&\leq \exp\left(-2\eta\sigma_{\min}(\mathbf{M})\,t\right)\sum_i\left(\sigma_i(\mathbf{U_0})^2 - \sigma_i(\mathbf{M})\right)^2 \\
&\leq \exp\left(-2\eta\sigma_{\min}(\mathbf{M})\,t\right)\left\|\mathbf{U_0}^2 - \mathbf{M}\right\|_F^2.
\end{aligned}$$

□

Note that the above proof crucially used the fact that the eigenvectors of $\mathbf{U_t}$ and $\mathbf{M}$ are *aligned*, to reduce the matrix iterations to iterations only over the singular values.

### 3.2 Approach

As we begin to investigate the global convergence properties of Eq.(4), the above argument breaks down due to lack of alignment between the singular vectors of $\mathbf{M}$ and those of the iterates $\mathbf{U_t}$. Let us now take a step back and consider non-convex optimization in general. There are two broad reasons why local search approaches fail for these problems. The first is the presence of local minima and the second is the presence of saddle points. Each of these presents different challenges: with local minima, local search approaches have no way of certifying whether the convergence point is a local minimum or global minimum; while with saddle points, if the iterates get close to a saddle point, the local neighborhood looks essentially flat and escaping the saddle point may take exponential time.

The starting point of our work is the realization that the non-convex formulation of the matrix squareroot problem does not have any local minima. This can be argued using the continuity of the matrix squareroot function, and this statement is indeed true for many matrix factorization problems. The only issue to be contended with is the presence of saddle points. In order to overcome this issue, it suffices to show that the iterates of the algorithm never get too close to a saddle point. More concretely, while optimizing a function $f$ with iterates $\mathbf{U_t}$, it suffices to show that for every $t$, $\mathbf{U_t}$ always stay in some region $\mathcal{D}$ far from saddle points so that for all $\mathbf{U}, \mathbf{U}' \in \mathcal{D}$:

$$\|\bigtriangledown f(\mathbf{U}) - \bigtriangledown f(\mathbf{U}')\|_F \leq L\|\mathbf{U} - \mathbf{U}'\|_F \tag{6}$$

$$\|\bigtriangledown f(\mathbf{U})\|_F \geq \sqrt{\ell\left(f(\mathbf{U}) - f_*\right)}, \tag{7}$$

where $f_* = \min_{\mathbf{U}} f(\mathbf{U})$, and $L$ and $\ell$ are some constants. If we flatten matrix $\mathbf{U}$ to be $n^2$-dimensional vector, then Eq.(6) is the standard smoothness assumption in optimization, and Eq.(7) is known as gradient dominated property [Polyak, 1963, Nesterov and Polyak, 2006]. If Eq.(6) and Eq.(7) hold, it follows from standard analysis that gradient descent with a step size $\eta < \frac{1}{L}$ achieves geometric convergence with

$$f(\mathbf{U_t}) - f_* \leq \exp\left(-\eta\ell t/2\right)\left(f(\mathbf{U_0}) - f_*\right).$$

Since the gradient in our setting is $\left(\mathbf{U_t}^2 - \mathbf{M}\right)\mathbf{U_t} + \mathbf{U_t}\left(\mathbf{U_t}^2 - \mathbf{M}\right)$, in order to establish Eq.(7), it suffices to lower bound $\lambda_{\min}(\mathbf{U_t})$. Similarly, in order to establish Eq.(6), it suffices to upper bound $\|\mathbf{U_t}\|_2$. Of course, we cannot hope to converge if we start from a saddle point. In particular Eq.(7) will not hold for any $l > 0$. The core of our argument consists of Lemmas 4.3 and 4.2, which essentially establish Eq.(6) and Eq.(7) respectively for the matrix squareroot problem Eq.(4), with the resulting parameters $l$ and $L$ dependent on the starting point $\mathbf{U_0}$. Lemmas 4.3 and 4.2 accomplish this by proving upper and lower bounds respectively on $\|\mathbf{U_t}\|_2$ and $\lambda_{\min}(\mathbf{U_t})$. The proofs of these lemmas use only elementary linear algebra and we believe such results should be possible for many more matrix factorization problems.

### 3.3 Guarantees

In this section, we will present our main results establishing that gradient descent on (4) converges to the matrix square root at a geometric rate and its robustness to errors in each iteration.

#### 3.3.1 Noiseless setting

The following theorem establishes geometric convergence of Algorithm 1 from a full rank initial point.

**Theorem 3.2.** *There exist universal numerical constants $c$ and $\widehat{c}$ such that if $\mathbf{U_0}$ is a PD matrix and $\eta < \frac{c}{\alpha\beta^2}$, then for every $t \in [T-1]$, we have $\mathbf{U_t}$ be a PD matrix with*

$$\left\|\mathbf{M} - \mathbf{U_t}^2\right\|_F \leq \exp\left(-\widehat{c}\eta\beta^2 t\right) \left\|\mathbf{M} - \mathbf{U_0}^2\right\|_F,$$

*where $\alpha$ and $\beta$ are defined as*

$$\alpha \triangleq \left(\frac{\max\left(\|\mathbf{U_0}\|_2, \sqrt{\|\mathbf{M}\|_2}\right)}{\min\left(\sigma_{\min}\left(\mathbf{U_0}\right), \sqrt{\sigma_{\min}\left(\mathbf{M}\right)}\right)}\right)^3,$$

$$\beta \triangleq \min\left(\sigma_{\min}\left(\mathbf{U_0}\right), \sqrt{\sigma_{\min}\left(\mathbf{M}\right)}\right)$$

**Remarks**:

- This result implies global geometric convergence. Choosing $\eta = \frac{c}{\alpha\beta^2}$, in order to obtain an accuracy of $\epsilon$, the number of iterations required would be $\mathcal{O}\left(\alpha \log \frac{\left\|\mathbf{M}-\mathbf{U_0}^2\right\|_F}{\epsilon}\right)$.
- Note that saddle points of (4) must be rank degenerate matrix ($\sigma_{\min}(\mathbf{U}) = 0$) and starting Algorithm 1 from a point close to the rank degenerate surface takes a long time to get away from the saddle surface. Hence, as $\mathbf{U_0}$ gets close to being rank degenerate, convergence rate guaranteed by Theorem 3.2 degrades (as $\kappa\left(\mathbf{U_0}\right)^3$). It is possible to obtain a smoother degradation with a finer analysis, but in the current paper, we trade off optimal results for a simple analysis.
- The convergence rate guaranteed by Theorem 3.2 also depends on the relative scales of $\mathbf{U_0}$ and $\mathbf{M}$ (say as measured by $\|\mathbf{U_0}\|_2^2 / \|\mathbf{M}\|_2$) and is best if it is close to 1.
- We believe that it is possible to extend our analysis to the case where $\mathbf{M}$ is low rank (PSD). In this case, suppose $\text{rank}(\mathbf{M}) = k$, and let $\mathbf{U}^\star$ be the k-dimensional subspace in which $\mathbf{M}$ resides. Then, saddle points should satisfy $\sigma_k(\mathbf{U}^\top\mathbf{U}^\star) = 0$.

A simple corollary of this result is when we choose $\mathbf{U_0} = \lambda\mathbf{I}$, where $\|\mathbf{M}\|_2 \leq \lambda \leq 2\|\mathbf{M}\|_2$ (such a $\lambda$ can be found in time $\mathcal{O}\left(n^2\right)$ Musco and Musco [2015]).

**Corollary 3.3.** *Suppose we choose $\mathbf{U_0} = \lambda\mathbf{I}$, where $\|\mathbf{M}\|_2 \leq \lambda \leq 2\|\mathbf{M}\|_2$. Then $\left\|\mathbf{M} - \mathbf{U_T}^2\right\|_F \leq \epsilon$ for $T \geq \mathcal{O}\left(\kappa^{\frac{3}{2}} \log \frac{\left\|\mathbf{M}-\mathbf{U_0}^2\right\|_F}{\epsilon}\right)$.*

#### 3.3.2 Noise Stability

Theorem 3.2 assumes that the gradient descent updates are performed with out any error. This is not practical. For instance, any implementation of Algorithm 1 would incur rounding errors. Our next result addresses this issue by showing that Algorithm 1 is stable in the presence of small, arbitrary errors in each iteration. This will establish the stability of our algorithm in the presence of round-off errors for instance. Formally, we consider in every gradient step, we incur an error $\triangle_\mathbf{t}$.

The following theorem shows that as long as the errors $\triangle_\mathbf{t}$ are small enough, Algorithm 1 recovers the true squareroot upto an accuracy of the error floor. The proof of the theorem follows fairly easily from that of Theorem 3.2.

**Theorem 3.4.** *There exist universal numerical constants $c$ and $\widehat{c}$ such that the following holds: Suppose $\mathbf{U_0}$ is a PD matrix and $\eta < \frac{c}{\alpha\beta^2}$ where $\alpha$ and $\beta$ are defined as before:*

$$\alpha \triangleq \left(\frac{\max\left(\|\mathbf{U_0}\|_2, \sqrt{\|\mathbf{M}\|_2}\right)}{\min\left(\sigma_{\min}\left(\mathbf{U_0}\right), \sqrt{\sigma_{\min}\left(\mathbf{M}\right)}\right)}\right)^3,$$

$$\beta \triangleq \min\left(\sigma_{\min}\left(\mathbf{U_0}\right), \sqrt{\sigma_{\min}\left(\mathbf{M}\right)}\right).$$

*Suppose further that $\|\triangle_\mathbf{t}\|_2 < \frac{1}{300}\eta\sigma_{\min}\left(\mathbf{M}\right)\beta$. Then, for every $t \in [T-1]$, we have $\mathbf{U_t}$ be a PD matrix with*

$$\left\|\mathbf{M} - \mathbf{U_t}^2\right\|_F \leq \exp\left(-\widehat{c}\eta\beta^2 t\right) \left\|\mathbf{M} - \mathbf{U_0}^2\right\|_F + 4\max(\|\mathbf{U_0}\|_2, \sqrt{3\|\mathbf{M}\|_2}) \sum_{s=0}^{t-1} e^{-\widehat{c}\eta\beta^2(t-s-1)} \|\triangle_\mathbf{s}\|_F.$$

**Remarks**:

- Since the errors above are multiplied by a decreasing sequence, they can be bounded to obtain a limiting accuracy of $\mathcal{O}\left(\alpha(\|\mathbf{U_0}\|_2 + \sqrt{\|\mathbf{M}\|_2})(\sup_s \|\triangle_\mathbf{s}\|_F)\right)$.
- If there is error in only the first iteration i.e., $\triangle_\mathbf{t} = 0$ for $t \neq 0$, then the initial error $\triangle_\mathbf{0}$ is attenuated with every iteration,

  $$\left\|\mathbf{M} - \mathbf{U_t}^2\right\|_F \leq \exp\left(-\widehat{c}\eta\beta^2 t\right) \left\|\mathbf{M} - \mathbf{U_0}^2\right\|_F + 6\max(\|\mathbf{U_0}\|_2^2, \|\mathbf{M}\|_2) e^{-\widehat{c}\eta\beta^2(t-1)} \|\triangle_\mathbf{0}\|_F.$$

  That is, our dependence on $\|\triangle_\mathbf{0}\|_F$ is exponentially decaying with respect to time $t$. On the

contrary, best known results only guarantees the error dependence on $\|\triangle_{\mathbf{0}}\|_F$ will not increase significantly with respect to time $t$ [Higham, 2008].

### 3.3.3 Lower Bound

We also prove the following lower bound showing that gradient descent with a fixed step size requires $\Omega(\kappa)$ iterations to achieve an error of $\mathcal{O}(\sigma_{\min}(\mathbf{M}))$.

**Theorem 3.5.** *For any value of $\kappa$, we can find a matrix $\mathbf{M}$ such that, for any step size $\eta$, there exists an initialization $\mathbf{U_0}$ that has the same eigenvectors as $\mathbf{M}$, with $\|\mathbf{U_0}\|_2 \leq \sqrt{3\|\mathbf{M}\|_2}$ and $\sigma_{\min}(\mathbf{U_0}) \geq \frac{1}{10}\sqrt{\sigma_{\min}(\mathbf{M})}$, such that we will have $\left\|\mathbf{U_t}^2 - \mathbf{M}\right\|_F \geq \frac{1}{4}\sigma_{\min}(\mathbf{M})$ for all $t \leq \kappa$.*

This lemma shows that the convergence rate of gradient descent fundamentally depends on the condition number $\kappa$, even if we start with a matrix that has the same eigenvectors and similar scale as $\mathbf{M}$. In this case, note that the lower bound of Theorem 3.5 is off from the upper bound of Theorem 3.2 by $\sqrt{\kappa}$. Though we do not elaborate in this paper, it is possible to formally show that a dependence of $\kappa^{3/2}$ is the best bound possible using our argument (i.e., one along the lines of Section 3.2).

## 4 Proof Sketch for Theorem 3.2

In this section, we will present the proof of Theorem 3.2. To make our strategy more concrete and transparent, we will leave the full proofs of some technical lemmas in Appendix A.

At a high level, our framework consists of following three steps:

1. Show all bad stationary points lie in a measure zero set $\{\mathbf{U}|\phi(\mathbf{U}) = 0\}$ for some constructed potential function $\phi(\cdot)$. In this paper, for the matrix squareroot problem, we choose the potential function $\phi(\cdot)$ to be the smallest singular value function $\sigma_{\min}(\cdot)$.
2. Prove for any $\epsilon > 0$, if initial $\mathbf{U}_0 \in \mathcal{D}_\epsilon = \{\mathbf{U}|\ |\phi(\mathbf{U})| > \epsilon\}$ and the stepsize is chosen appropriately, then we have all iterates $\mathbf{U}_t \in \mathcal{D}_\epsilon$. That is, updates will always keep away from bad stationary points.
3. Inside regions $\mathcal{D}_\epsilon$, show that the optimization function satisfies good properties such as smoothness and gradient-dominance, which establishes convergence to a global minimum with good rate.

Since we can make $\epsilon$ arbitrarily small and since $\{\mathbf{U}|\phi(\mathbf{U}) = 0\}$ is a measure zero set, this essentially establishes convergence from a (Lebesgue) measure one set, proving global convergence.

We note that step 2 above implies that no stationary point found in the set $\{\mathbf{U}|\phi(\mathbf{U}) = 0\}$ is a local minimum – it must either be a saddle point or a local maximum. This is because starting at any point outside $\{\mathbf{U}|\phi(\mathbf{U}) = 0\}$ does not converge to $\{\mathbf{U}|\phi(\mathbf{U}) = 0\}$. Therefore, our framework can be mostly used for non-convex problems with saddle points but no spurious local minima.

Before we proceed with the full proof, we will first illustrate the three steps above for a simple, special case where $n = 2$ and all relevant matrices are diagonal. Specifically, we choose target matrix $\mathbf{M}$ and parameterize $\mathbf{U}$ as:

$$\mathbf{M} = \begin{pmatrix} 4 & 0 \\ 0 & 2 \end{pmatrix}, \qquad \mathbf{U} = \begin{pmatrix} x & 0 \\ 0 & y \end{pmatrix}.$$

Here $x$ and $y$ are unknown parameters. Since we are concerned with $\mathbf{U} \succeq 0$, we see that $x, y \geq 0$. The reason we restrict ourselves to diagonal matrices is so that the parameter space is two dimensional letting us give a good visual representation of the parameter space. Figures 1 and 2 show the plots of function value contours and negative gradient flow respectively as a function of $x$ and $y$.

We will use Figures 1 and 2 to qualitatively establish the three steps in our framework.

1. From Figure 1, we note that $(2, \sqrt{2})$ is the global minimum. $(2, 0), (0, \sqrt{2})$ are saddle points, while $(0, 0)$ is local maximum. We notice all the stationary points which are not global minima lie on the surface $\sigma_{\min}(\mathbf{U}) = 0$, that is, the union of x-axis and y-axis.
2. By defining a boundary $\{\mathbf{U}|\sigma_{\min}(\mathbf{U}) > c, \|\mathbf{U}\|_2 < C\}$ for some small $c$ and large $C$ (corresponding to the red box in Figure 2), we see that negative gradient flow is pointed inside the box which means that for any point in the box, performing gradient descent with a small enough stepsize will ensure that all iterates lie inside the box (and hence keep away from saddle points).
3. Inside the red box, Figure 2 shows that negative gradient flow points to the global optimum. Moreover, we can indeed establish upper and lower bounds on the magnitude of gradients within the red box – this corresponds to establishing smoothness and gradient dominance respectively.

Together, all the above observations along with standard results in optimization tell us that gradient descent has geometric convergence for this problem.

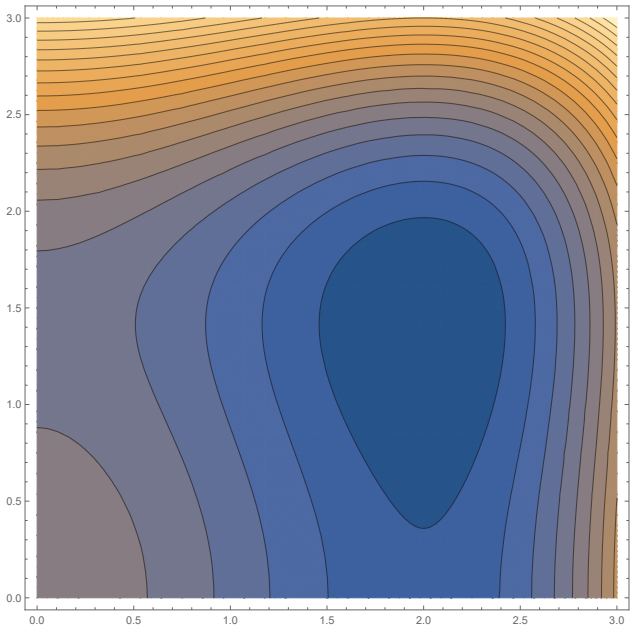

Figure 1: Contour of Objective Functions

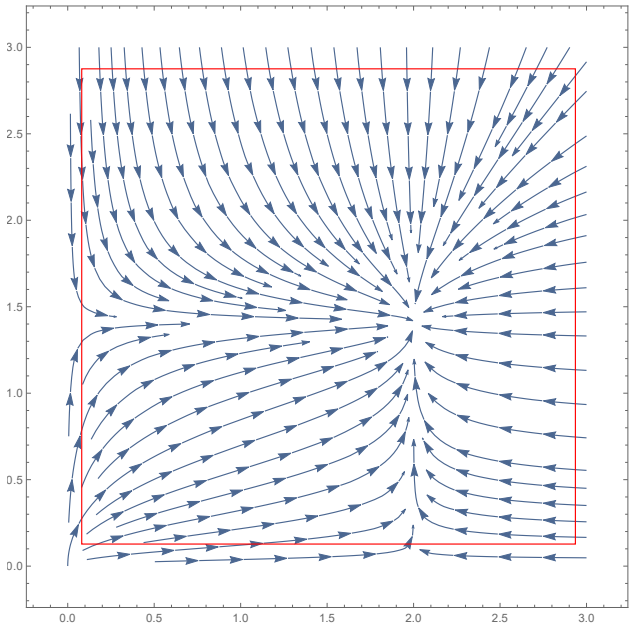

Figure 2: Flow of Negative Gradient

We now present a formal proof of our result.

### 4.1 Location of Saddle Points

We first give a characterization of locations of all the stationary points which are not global minima.

**Lemma 4.1.** *Within symmetric PSD cone* $\{\mathbf{U}|\mathbf{U} \succeq 0\}$*, all stationary points of* $f(\mathbf{U}) = \left\|\mathbf{M} - \mathbf{U}^2\right\|_F^2$ *which are not global minima, must satisfy* $\sigma_{\min}(\mathbf{U}) = 0$*.*

*Proof.* For any stationary point $\mathbf{U}'$ of $f(\mathbf{U})$ which is not on the boundary $\{\mathbf{U}|\sigma_{\min}(\mathbf{U}) = 0\}$, by linear algebra calculation, we have:

$$\begin{aligned}0 =&\|\nabla f(\mathbf{U}')\|_F^2 = \|(\mathbf{U}'^2 - \mathbf{M})\mathbf{U}' + \mathbf{U}'(\mathbf{U}'^2 - \mathbf{M})\|_F^2\\ =&\langle(\mathbf{U}'^2 - \mathbf{M})\mathbf{U}' + \mathbf{U}'(\mathbf{U}'^2 - \mathbf{M}),\\ &\qquad(\mathbf{U}'^2 - \mathbf{M})\mathbf{U}' + \mathbf{U}'(\mathbf{U}'^2 - \mathbf{M})\rangle\\ =&2\mathrm{Tr}([(\mathbf{U}'^2 - \mathbf{M})\mathbf{U}']^2) + 2\mathrm{Tr}((\mathbf{U}'^2 - \mathbf{M})^2\mathbf{U}'^2)\\ \geq&4\sigma_{\min}^2(\mathbf{U}')\|\mathbf{U}'^2 - \mathbf{M}\|_F^2\end{aligned}$$

Therefore, since $\mathbf{U}'$ is not on the boundary of PSD cone, we have $\sigma_{\min}^2(\mathbf{U}) > 0$, which gives $f(\mathbf{U}') = \left\|\mathbf{M} - \mathbf{U}'^2\right\|_F^2 \neq 0$, thus $\mathbf{U}'$ is global minima. □

As mentioned before, note that all the bad stationary points are contained in $\{\mathbf{U}|\sigma_{\min}(\mathbf{U}) = 0\}$ which is a (Lebesgue) measure zero set.

### 4.2 Stay Away from Saddle Surface

Since the gradient at stationary points is zero, gradient descent can never converge to a global minimum if starting from suboptimal stationary points. Fortunately, in our case, gradient updates will keep away from bad stationary points. As in next Lemma, we show that as long as we choose suitable small learning rate, $\sigma_{\min}(\mathbf{U}_t)$ will never be too small.

**Lemma 4.2.** *Suppose* $\eta < c\frac{\min\left(\sigma_{\min}(\mathbf{U_0}), \sigma_{\min}^{1/2}(\mathbf{M})/10\right)}{\max\left(\|\mathbf{U_0}\|_2^3, \left(3\|\mathbf{M}\|_2\right)^{3/2}\right)}$*, where* $c$ *is a small enough constant. Then, for every* $t \in [T-1]$*, we have* $\mathbf{U_t}$ *in Algorithm 1 be a PD matrix with*

$$\lambda_{\min}(\mathbf{U_t}) \geq \min\left(\sigma_{\min}(\mathbf{U_0}), \frac{\sqrt{\sigma_{\min}(\mathbf{M})}}{10}\right).$$

It turns out that the gradient updates will not only keep $\sigma_{\min}(\mathbf{U})$ from being too small, but also keep $\|\mathbf{U}\|_2$ from being too large.

**Lemma 4.3.** *Suppose* $\eta < \frac{1}{10\max\left(\|\mathbf{U_0}\|_2^2, 3\|\mathbf{M}\|_2\right)}$*. For every* $t \in [T-1]$*, we have* $\mathbf{U_t}$ *in Algorithm 1 satisfying:*

$$\|\mathbf{U_t}\|_2 \leq \max\left(\|\mathbf{U_0}\|_2, \sqrt{3\|\mathbf{M}\|_2}\right).$$

Although $\|\mathbf{U}\|_2$ is not directly related to the surface with bad stationary points, the upper bound on $\|\mathbf{U}\|_2$ is crucial for the smoothness of function $f(\cdot)$, which gives good convergence rate in Section 4.3.

### 4.3 Convergence in Saddle-Free Region

So far, we have been able to establish both upper bounds and lower bounds on singular values of all iterates $\mathbf{U}_t$ given suitable small learning rate. Next, we show that when spectral norm of $\mathbf{U}$ is small, function $f(\mathbf{U})$ is smooth, and when $\sigma_{\min}(\mathbf{U})$ is large, function $f(\mathbf{U})$ is gradient dominated.

**Lemma 4.4.** *Function* $f(\mathbf{U}) = \left\|\mathbf{M}-\mathbf{U}^2\right\|_F^2$ *is* $8\max\{\Gamma, \|\mathbf{M}\|_2\}$*-smooth in region* $\{\mathbf{U}|\,\|\mathbf{U}\|_2^2 \le \Gamma\}$*. That is, for any* $\mathbf{U_1}, \mathbf{U_2} \in \{\mathbf{U}|\,\|\mathbf{U}\|_2^2 \le \Gamma\}$*, we have:*

$$\|\nabla f(\mathbf{U_1}) - \nabla f(\mathbf{U_2})\|_F \le 8\max\{\Gamma, \|\mathbf{M}\|_2\}\|\mathbf{U_1}-\mathbf{U_2}\|_F$$

**Lemma 4.5.** *Function* $f(\mathbf{U}) = \left\|\mathbf{M}-\mathbf{U}^2\right\|_F^2$ *is* $4\gamma$*-gradient dominated in region* $\{\mathbf{U}|\sigma_{\min}(\mathbf{U})^2 \ge \gamma\}$*. That is, for any* $\mathbf{U} \in \{\mathbf{U}|\sigma_{\min}(\mathbf{U})^2 \ge \gamma\}$*, we have:*

$$\|\nabla f(\mathbf{U})\|_F^2 \ge 4\gamma f(\mathbf{U})$$

Lemma 4.4 and 4.5 are the formal versions of Eq.(6) and Eq.(7) in Section 3.2, which are essential in establishing geometric convergence.

Putting all pieces together, we are now ready prove our main theorem:

*Proof of Theorem 3.2.* Recall the definitions in Theorem 3.2:

$$\alpha \triangleq \left(\frac{\max\left(\|\mathbf{U_0}\|_2, \sqrt{\|\mathbf{M}\|_2}\right)}{\min\left(\sigma_{\min}(\mathbf{U_0}), \sqrt{\sigma_{\min}(\mathbf{M})}\right)}\right)^3,$$
$$\beta \triangleq \min\left(\sigma_{\min}(\mathbf{U_0}), \sqrt{\sigma_{\min}(\mathbf{M})}\right)$$

By choosing learning rate $\eta < \frac{c}{\alpha\beta^2}$ with small enough constant $c$. We can satisfy the precondition of Lemma 4.2, and Lemma 4.3 at same time. Therefore, we know all iterates will fall in region:

$$\left\{\mathbf{U}\,\middle|\,\|\mathbf{U}\|_2 \le \max\left(\|\mathbf{U_0}\|_2, \sqrt{3\|\mathbf{M}\|_2}\right),\ \lambda_{\min}(\mathbf{U}) \ge \min\left(\sigma_{\min}(\mathbf{U_0}), \frac{\sqrt{\sigma_{\min}(\mathbf{M})}}{10}\right)\right\}$$

Then, apply Lemma 4.4 and Lemma 4.5, we know in this region, function $f(\mathbf{U}) = \left\|\mathbf{U}^2-\mathbf{M}\right\|_F^2$ has smoothness parameter:

$$8\max\left\{\max\left\{\|\mathbf{U_0}\|_2^2, 3\|\mathbf{M}\|_2\right\}, \|\mathbf{M}\|_2\right\} \le 24\alpha^{2/3}\beta^2$$

and gradient dominance parameter:

$$4\min\left\{\sigma_{\min}^2(\mathbf{U_0}), \frac{\sigma_{\min}(\mathbf{M})}{100}\right\} \ge \frac{\beta^2}{25}$$

That is, $f(\mathbf{U})$ in the region is both $24\alpha^{2/3}\beta^2$-smooth, and $\beta^2/25$-gradient dominated.

Finally, by Taylor's expansion of smooth function, we have:

$$\begin{aligned} f(\mathbf{U_{t+1}}) \le& f(\mathbf{U_t}) + \langle\nabla f(\mathbf{U_t}), \mathbf{U_{t+1}} - \mathbf{U_t}\rangle \\ &+ 12\alpha^{2/3}\beta^2\|\mathbf{U_{t+1}} - \mathbf{U_t}\|_F^2 \\ =& f(\mathbf{U_t}) - (\eta - 12\eta^2\alpha^{2/3}\beta^2)\|\nabla f(\mathbf{U_t})\|_F^2 \\ \le& f(\mathbf{U_t}) - \frac{\eta}{2}\|\nabla f(\mathbf{U_t})\|_F^2 \\ \le& (1-\eta\frac{\beta^2}{50})f(\mathbf{U_t}) \end{aligned}$$

The second last inequality is again by setting constant $c$ in learning rate to be small enough, and the last inequality is by the property of gradient dominated. This finishes the proof.

□

## 5 Conclusion

In this paper, we take a first step towards addressing the large gap between local convergence results with good convergence rates and global convergence results with highly suboptimal convergence rates. We consider the problem of computing the squareroot of a PD matrix, which is a widely studied problem in numerical linear algebra, and show that non-convex gradient descent achieves global geometric convergence with a good rate. In addition, our analysis also establishes the stability of this method to numerical errors. We note that this is the first method to have provable robustness to numerical errors for this problem and our result illustrates that global convergence results are also useful in practice since they might shed light on the stability of optimization methods.

Our result shows that even in the presence of a large saddle point surface, gradient descent might be able to avoid it and converge to the global optimum at a linear rate. We believe that our framework and proof techniques should be applicable for several other non-convex problems (especially those based on matrix factorization) in machine learning and numerical linear algebra and would lead to the analysis of gradient descent and stochastic gradient descent in a transparent way while also addressing key issues like robustness to noise or numerical errors.

## References


Alekh Agarwal, Animashree Anandkumar, Prateek Jain, and Praneeth Netrapalli. Learning sparsely used overcomplete dictionaries via alternating minimization. *arXiv preprint*, 2013.

Michal Aharon, Michael Elad, and Alfred Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 54(11):4311–4322, 2006.

Åke Björck and Sven Hammarling. A Schur method for the square root of a matrix. *Linear algebra and its applications*, 52:127–140, 1983.

Emmanuel J Candes, Xiaodong Li, and Mahdi Soltanolkotabi. Phase retrieval via wirtinger flow: Theory and algorithms. *Information Theory, IEEE Transactions on*, 61(4):1985–2007, 2015.

Neal Carlson. Federated square root filter for decentralized parallel processors. *Aerospace and Electronic Systems, IEEE Transactions on*, 26(3):517–525, 1990.

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

Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

Nicholas J Higham. Newtons method for the matrix square root. *Mathematics of Computation*, 46(174): 537–549, 1986.

Nicholas J Higham. Computing real square roots of a real matrix. *Linear Algebra and its applications*, 88: 405–430, 1987.

Nicholas J Higham. Stable iterations for the matrix square root. *Numerical Algorithms*, 15(2):227–242, 1997.

Nicholas J Higham. *Functions of matrices: theory and computation*. Society for Industrial and Applied Mathematics (SIAM), 2008.

Prateek Jain and Praneeth Netrapalli. Fast exact matrix completion with finite samples. *arXiv preprint*, 2014.

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

Paul G Kaminski, Arthur E Bryson Jr, and Stanley F Schmidt. Discrete square root filtering: A survey of current techniques. *Automatic Control, IEEE Transactions on*, 16(6):727–736, 1971.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8):30–37, 2009.

Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.

Jason D Lee, Max Simchowitz, Michael I Jordan, and Benjamin Recht. Gradient descent converges to minimizers. *University of California, Berkeley*, 1050:16, 2016.

Beatrice Meini. The matrix square root from a new functional perspective: theoretical results and computational issues. *SIAM journal on matrix analysis and applications*, 26(2):362–376, 2004.

Cameron Musco and Christopher Musco. Stronger approximate singular value decomposition via the block lanczos and power methods. *arXiv preprint*, 2015.

Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

Praneeth Netrapalli, Prateek Jain, and Sujay Sanghavi. Phase retrieval using alternating minimization. *IEEE Transactions on Signal Processing*, 63 (18):4814–4826, 2015.

Victor Y Pan, Zhao Q Chen, and Ailong Zheng. The complexity of the algebraic eigenproblem. *Mathematical Sciences Research Institute, Berkeley*, page 71, 1998.

BT Polyak. Gradient methods for the minimisation of functionals. *USSR Computational Mathematics and Mathematical Physics*, 3(4):864–878, 1963.

Benjamin Recht, Maryam Fazel, and Pablo A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

Suvrit Sra. On the matrix square root via geometric optimization. *arXiv preprint*, 2015.

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

Ruoyu Sun and Zhi-Quan Luo. Guaranteed matrix completion via nonconvex factorization. In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pages 270–289. IEEE, 2015.

Michael K Tippett, Jeffrey L Anderson, Craig H Bishop, Thomas M Hamill, and Jeffrey S Whitaker.

Ensemble square root filters*. *Monthly Weather Review*, 131(7):1485–1490, 2003.

Stephen Tu, Ross Boczar, Mahdi Soltanolkotabi, and Benjamin Recht. Low-rank solutions of linear matrix equations via procrustes flow. *arXiv preprint arXiv:1507.03566*, 2015.

Ronell Van Der Merwe and Eric Wan. The square-root unscented kalman filter for state and parameter-estimation. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 6, pages 3461–3464. IEEE, 2001.

Virginia Vassilevska Williams. Multiplying matrices faster than coppersmith-winograd. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 887–898. ACM, 2012.