Low-rank Matrix Completion using Alternating Minimization

[Extended Abstract]

Prateek Jain
Microsoft Research India, Bangalore
prajain@microsoft.com

Praneeth Netrapalli *
The University of Texas at Austin
praneethn@utexas.edu

Sujay Sanghavi
The University of Texas at Austin
sanghavi@mail.utexas.edu

ABSTRACT

Alternating minimization represents a widely applicable and empirically successful approach for finding low-rank matrices that best fit the given data. For example, for the problem of low-rank matrix completion, this method is believed to be one of the most accurate and efficient, and formed a major component of the winning entry in the Netflix Challenge [17].

In the alternating minimization approach, the low-rank target matrix is written in a *bi-linear form*, i.e. $X = UV^\dagger$; the algorithm then alternates between finding the best U and the best V. Typically, each alternating step in isolation is convex and tractable. However the overall problem becomes non-convex and is prone to local minima. In fact, there has been almost no theoretical understanding of when this approach yields a good result.

In this paper we present one of the first theoretical analyses of the performance of alternating minimization for matrix completion, and the related problem of matrix sensing. For both these problems, celebrated recent results have shown that they become well-posed and tractable once certain (now standard) conditions are imposed on the problem. We show that alternating minimization *also* succeeds under similar conditions. Moreover, compared to existing results, our paper shows that alternating minimization guarantees faster (in particular, geometric) convergence to the true matrix, while allowing a significantly simpler analysis.

Categories and Subject Descriptors

F.2.1 [**Numerical Algorithms and Problems**]: Computation on matrices

Keywords

Matrix Completion, Alternating Minimization

*Part of this work was done while the author was an intern at Microsoft Research India, Bangalore.

STOC'13, June 1-4, 2013, Palo Alto, California, USA.

1. INTRODUCTION

Finding a low-rank matrix to fit / approximate observations is a fundamental task in data analysis. In a slew of applications, a popular empirical approach has been to represent the target rank k matrix $X \in \mathbb{R}^{m\times n}$ in a *bi-linear form* $X = UV^\dagger$, where $U \in \mathbb{R}^{m\times k}$ and $V \in \mathbb{R}^{n\times k}$. Typically, this is done for two reasons:
(a) Size and computation: If the rank k of the target matrix (to be estimated) is much smaller than m, n, then U, V are significantly smaller than X and hence are more efficient to optimize for. This is crucial for several practical applications, e.g., recommender systems where one routinely encounters matrices with billions of entries.
(b) Modeling: In several applications, one would like to impose extra constraints on the target matrix, besides just low rank. Oftentimes, these constraints might be easier and more natural to impose on factors U, V. For example, in Sparse PCA [25], one looks for a low-rank X that is the product of *sparse* U and V.

Due to the above two reasons, in several applications, the target matrix X is parameterized by $X = UV^\dagger$. For example, clustering [16], sparse PCA [25] etc.

Using the bi-linear parametrization of the target matrix X, the task of estimating X now reduces to finding U and V that, for example, minimize an error metric. The resulting problem is typically non-convex due to bi-linearity. Correspondingly, a popular approach has been to use **alternating minimization:** iteratively keep one of U, V fixed and optimize over the other, then switch and repeat, see e.g. [18]. While the overall problem is non-convex, each sub-problem is typically convex and can be solved efficiently.

Despite the wide usage of bi-linear representation and alternating minimization, global optimality guarantees for such methods are still lacking.Motivated by this disconnect between theory and practice in the estimation of low-rank matrices, in this paper we provide **one of the first guarantees for performance of alternating minimization**, for two low-rank matrix recovery problems: matrix completion, and matrix sensing.

Matrix completion involves completing a low-rank matrix, by observing only a few of its elements. Its recent popularity, and primary motivation, comes from recommendation systems [18], where the task is to complete a user-item ratings matrix using only a small number of ratings. As elaborated in Section 2, alternating minimization becomes particularly appealing for this problem as it provides a fast, distributed

algorithm that can exploit both sparsity of ratings as well as the low-rank bi-linear parametrization of X.

Matrix sensing refers to the problem of recovering a low-rank matrix $M \in \mathbb{R}^{m\times n}$ from affine equations. That is, given d linear measurements $b_i = tr(A_i^\dagger M)$ and the corresponding measurement matrices A_i's, the goal is to recover back M. This problem is particularly interesting in the case of $d \ll mn$ and was first studied in [22] and subsequently in [10, 19]. In fact, matrix completion is a special case of this problem, where each observed entry in the matrix completion problem represents one single-element measurement matrix A_i.

Without any extra conditions, both matrix sensing and matrix completion are ill-posed problems, with potentially multiple low-rank solutions, and are in general NP hard [20, 21]. Current work on these problems thus impose some extra conditions, which makes the problems both well defined, and amenable to solution via the respective proposed algorithms [22, 3]; see Section 3 for more details. In this paper, we show that **under similar conditions to the ones used by the existing methods,** alternating minimization also guarantees recovery of the true matrix; we also show that it requires only a small number of computationally cheap iterations and hence, as observed empirically, is computationally much more efficient than the existing methods.

Notations: We represent a matrix by capital letter (e.g. M) and a vector by small letter (u). u_i represents i-th element of u and U_{ij} denotes (i,j)-th entry of U. U_i represents i-th column of U and $U^{(i)}$ represents i-th row of U. $A^\dagger$ denotes matrix transpose of A. $u = vec(U)$ represents vectorized U, i.e., $u = [U_1^\dagger\ U_2^\dagger\ \dots\ U_k^\dagger]^\dagger$. $\|u\|_p$ denotes L_p norm of u, i.e., $\|u\|_p = (\sum_i |u_i|^p)^{1/p}$. By default, $\|u\|$ denotes L_2 norm of u. $\|A\|_F$ denotes Frobenius norm of A, i.e., $\|vec(A)\|_2$. $\|A\|_2 = \max_{x,\|x\|_2=1} \|Ax\|_2$ denotes spectral norm of A. $tr(A)$ denotes the trace (sum of diagonal elements) of square matrix A. Typically, $\widehat{U}$, $\widehat{V}$ represent factor matrices (i.e., $\widehat{U} \in \mathbb{R}^{m\times k}$ and $\widehat{V} \in \mathbb{R}^{n\times k}$) and U, V represent their orthonormal basis.

Due to lack of space, we ommit some of the proofs; please refer [11] for detailed proofs.

2. OUR RESULTS

In this section, we will first define the matrix sensing problem, and present our results for it. Subsequently, we will do the same for matrix completion. The matrix sensing setting – i.e. recovery of any low-rank matrix from linear measurements that satisfy matrix RIP – represents an easier analytical setting than matrix completion, but still captures several key properties of the problem that helps us in developing an analysis for matrix completion.We note that for either problem, ours represent the first global optimality guarantees for alternating minimization based algorithms. [1]

Matrix Sensing via Alternating Minimization

Given d linear measurements $b_i = \langle M, A_i\rangle = tr(A_i^\dagger M)$, $1 \le i \le d$ of an *unknown* rank-k matrix $M \in \mathbb{R}^{m\times n}$ and the sensing matrices $A_i, 1 \le i \le d$, the goal in matrix sensing is to recover back M. In the following we collate these coefficients, so that $b \in \mathbb{R}^d$ is the vector of b_i's, and $\mathcal{A}(\cdot) : \mathbb{R}^{m\times n} \to d$ is the corresponding linear map, with $b = \mathcal{A}(M)$. With this notation, the Low-Rank Matrix Sensing problem is:

$$\text{Find } X \in \mathbb{R}^{m\times n},\ \ s.t\ \ \mathcal{A}(X) = b,\ \ \ rank(X) \le k. \quad \text{(LRMS)}$$

As in the existing work [22] on this problem, we are interested in the **under-determined case**, where $d < mn$. Note that this problem is a strict generalization of the popular compressed sensing problem [4]; compressed sensing represents the case when M is restricted to be a diagonal matrix.

For matrix sensing, alternating minimization approach involves representing X as a product of two matrices $U \in \mathbb{R}^{m\times k}$ and $V \in \mathbb{R}^{n\times k}$, i.e., $X = UV^\dagger$. If k is (much) smaller than m, n, then these matrices will be (much) smaller than X. With this bi-linear representation, alternating minimization can be viewed as an approximate way to solve the following non-convex optimization problem:

$$\min_{U\in\mathbb{R}^{m\times k}, V\in\mathbb{R}^{n\times k}} \quad \|\mathcal{A}(UV^\dagger) - b\|_2^2$$

As mentioned earlier, the alternating minimization algorithm for matrix sensing now alternately solves for U and V while fixing the other factor. See Algorithm 1 for a pseudo-code of the AltMinSense algorithm that we analyze.

We note two key properties of AltMinSense : a) Each minimization – over U with V fixed, and vice versa – is a simple least-squares problem, which can be solved in time $O(dn^2k^2 + n^3k^3)$[2], b) We initialize U^0 to be the top-k left singular vectors of $\sum_i A_i b_i$ (step 2 of Algorithm 1). As we will see later in Section 4, this provides a good initialization point for the sensing problem which is crucial; if the first iterate $\widehat{U}^0$ is orthogonal, or almost orthogonal, to the true U^* subspace, AltMinSense may never converge to the true space (this is easy to see in the simplest case, when the map is identity, i.e. $\mathcal{A}(X) = X$ – in which case AltMinSense just becomes the power method).

In general, since $d < mn$, problem (LRMS) is not well posed as there can be multiple rank-k solutions that satisfy $\mathcal{A}(X) = b$. However, inspired by a similar condition in compressed sensing [4], Recht et al. [22] showed that if the linear map $\mathcal{A}$ satisfies a *(matrix) restricted isometry property* (RIP), then a trace-norm based convex relaxation of (LRMS) leads to exact recovery. This property is defined below.

DEFINITION 2.1. *[22] A linear operator $\mathcal{A}(\cdot) : \mathbb{R}^{m\times n} \to \mathbb{R}^d$ is said to satisfy k-RIP, with δ_k RIP constant, if for all $X \in \mathbb{R}^{m\times n}$ s.t. $rank(X) \le k$, the following holds:*

$$(1-\delta_k)\|X\|_F^2 \le \|\mathcal{A}(X)\|_2^2 \le (1+\delta_k)\|X\|_F^2\,. \qquad (1)$$

Several random matrix ensembles with sufficiently many measurements (d) satisfy matrix RIP [22]. For example, if $d = \Omega(\frac{1}{\delta_k^2} kn \log n)$ and each entry of A_i is sampled i.i.d. from

Algorithm 1 AltMinSense : Alternating minimization for matrix sensing

1: Input $b, \mathcal{A}$
2: Initialize $\widehat{U}^0$ to be the top-k left singular vectors of $\sum_i A_i b_i$
3: **for** $t = 0, \cdots, T-1$ **do**
4: $\quad \widehat{V}^{t+1} \leftarrow \text{argmin}_{V\in\mathbb{R}^{n\times k}}\ \|\mathcal{A}(\widehat{U}^t V^\dagger) - b\|_2^2$
5: $\quad \widehat{U}^{t+1} \leftarrow \text{argmin}_{U\in\mathbb{R}^{m\times k}}\ \|\mathcal{A}(U\,(\widehat{V}^{t+1})^\dagger) - b\|_2^2$
6: **end for**
7: Return $X = \widehat{U}^T(\widehat{V}^T)^\dagger$

[1] Independent of our work, [13] also proved similar result for the matrix completion problem. See Section 3 for a detailed comparision of our results with that of [13].

[2] Throughout this paper, we assume $m \le n$.

a 0-mean sub-Gaussian distribution then k-RIP is satisfied with RIP constant δ_k.

We now present our main result for AltMinSense.

THEOREM 2.2. *Let $M = U^*\Sigma^*V^{*\dagger}$ be a rank-k matrix with non zero singular values $\sigma_1^* \geq \sigma_2^* \cdots \geq \sigma_k^*$. Also, let the linear measurement operator $\mathcal{A}(\cdot) : \mathbb{R}^{m\times n} \to \mathbb{R}^d$ satisfy $2k$-RIP with RIP constant $\delta_{2k} < \frac{(\sigma_k^*)^2}{(\sigma_1^*)^2}\frac{1}{100k}$. Then, in the AltMinSense algorithm (Algorithm 1), for all $T > 2\log(\|M\|_F/\epsilon)$, the iterates $\widehat{U}^T$ and $\widehat{V}^T$ satisfy:*

$$\|M - \widehat{U}^T(\widehat{V}^T)^\dagger\|_F \leq \epsilon.$$

The above theorem establishes geometric convergence (in $O(\log(1/\epsilon))$ steps) of AltMinSense to the optimal solution of (LRMS) under standard RIP assumptions. In contrast, all the existing iterative methods for trace-norm minimization require at least $O(\frac{1}{\sqrt{\epsilon}})$ steps; interior point methods for trace-norm minimization converge to the optimum in $O(\log(1/\epsilon))$ steps but require storage of the full $m\times n$ matrix and require $O(n^5)$ time per step, which makes it infeasible for even moderate sized problems.

Recently, several projected gradient based methods have been developed for matrix sensing [10, 19] that also guarantee convergence to the optimum in $O(\log(1/\epsilon))$ steps. But each iteration in these algorithms requires computation of the top k singular components of an $m \times n$ matrix, which is typically significantly slower than solving a least squares problem (as required by each iteration of AltMinSense).

Stagewise AltMinSense Algorithm: A drawback of our analysis for AltMinSense is the dependence of δ_{2k} on the condition number ($\kappa = \frac{\sigma_1^*}{\sigma_k^*}$) of M, which implies that the number of measurements d required by AltMinSense grows quadratically with κ. We address this issue by using a stagewise version of AltMinSense (Algorithm 3) for which we are able to obtain near optimal measurement requirement.

The key idea behind our stagewise algorithm is that if one of the singular vectors of M is very dominant, then we can treat the underlying matrix as a rank-1 matrix plus noise and approximately recover the top singular vector. Once we remove this singular vector from the measurements, we will have a relatively well-conditioned problem. Hence, at each stage of Algorithm 3, we seek to remove the remaining most dominant singular vector of M. See Section 6 for more details; here we state the corresponding theorem regarding the performance of Stage-AltMin.

THEOREM 2.3. *Let $M = U^*\Sigma^*V^{*\dagger}$ be a rank-k incoherent matrix with non zero singular values $\sigma_1^* \geq \sigma_2^* \cdots \geq \sigma_k^*$. Also, let $\mathcal{A}(\cdot) : \mathbb{R}^{m\times n} \to \mathbb{R}^d$ be a linear measurement operator that satisfies $2k$-RIP with RIP constant $\delta_{2k} < \frac{1}{3200k^2}$. Suppose, Stage-AltMin (Algorithm 3) is supplied inputs $\mathcal{A}$, $b = \mathcal{A}(M)$. Then, the i-th stage iterates $\widehat{U}_{1:i}^T$, $V_{1:i}^T$ satisfy:*

$$\|M - \widehat{U}_{1:i}^T\left(V_{1:i}^T\right)^\dagger\|_F^2 \leq \max(\epsilon, 16k(\sigma_{i+1}^*)^2),$$

where $T = \Omega\left(\log(\|M\|_F^2/\epsilon)\right)$. That is, the T-th step iterates of the k-th stage, satisfy: $\|M - \widehat{U}_{1:k}^T\left(V_{1:k}^T\right)^\dagger\|_F^2 \leq \epsilon$.

The above theorem guarantees exact recovery using $O(k^4 n \log n)$ measurements which is only $O(k^3)$ worse than the information theoretic lower bound. We also note that for simplicity of analysis, we did not optimize the constant factors in δ_{2k}.

Matrix Completion via Alternating Minimization

The matrix completion problem is the following: there is an unknown rank-k matrix $M \in \mathbb{R}^{m\times n}$, of which we know a set $\Omega \subset [m] \times [n]$ of elements; that is, we know the values of elements M_{ij}, for $(i,j) \in \Omega$. The task is to recover M. Formally, the Low-Rank Matrix Completion problem is:

$$\text{Find rank-}k \text{ matrix } X \text{ s.t. } P_\Omega(X) = P_\Omega(M), \quad \text{(LRMC)}$$

where for any matrix S and a set of elements $\Omega \subset [m]\times[n]$ the matrix $P_\Omega(S) \in \mathbb{R}^{m\times n}$ is as defined below:

$$P_\Omega(S)_{ij} = \begin{cases} S_{ij} & \text{if } (i,j) \in \Omega, \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

We are again interested in the under-determined case; in fact, for a fixed rank k, as few as $O(n \log n)$ elements may be observed. This problem is a special case of matrix sensing, with the measurement matrices $A_i = e_j e_\ell^\dagger$ being non-zero only in single elements; however, such matrices do not satisfy matrix RIP conditions like (1). For example, consider a low-rank $M = e_1 e_1^\dagger$ for which a uniformly random Ω of size $O(n \log n)$ will most likely miss the non-zero entry of M.

Nevertheless, like matrix sensing, matrix completion has been shown to be possible once additional conditions are applied to the low-rank matrix M and the observation set Ω. Starting with the first work [3], the typical assumption has been to have Ω generated uniformly at random, and M to satisfy a particular incoherence property that, loosely speaking, makes it very far from a sparse matrix. In this paper, we show that *once* such assumptions are made, alternating minimization *also* succeeds. We now restate, and subsequently use, this incoherence definition.

DEFINITION 2.4. *[3] A matrix $M \in \mathbb{R}^{m\times n}$ is incoherent with parameter μ if:*

$$\left\|u^{(i)}\right\|_2 \leq \frac{\mu\sqrt{k}}{\sqrt{m}} \; \forall\, i \in [m], \;\; \left\|v^{(j)}\right\|_2 \leq \frac{\mu\sqrt{k}}{\sqrt{n}} \; \forall\, j \in [n], \quad (3)$$

where $M = U\Sigma V^T$ is the SVD of M and $u^{(i)}$, $v^{(j)}$ denote the i^{th} row of U and the j^{th} row of V respectively.

The alternating minimization algorithm can be viewed as an approximate way to solve the following non-convex problem:

$$\min_{U,V\in\mathbb{R}^{n\times k}} \quad \|P_\Omega(UV^\dagger) - P_\Omega(M)\|_F^2$$

Similar to AltMinSense, the altmin procedure proceeds by alternatively solving for U and V. As noted earlier, this approach has been popular in practice and has seen several variants and extensions being used in practice [24, 18, 7]. However, for ease of analysis, our algorithm further modifies the standard alternating minimization method. In particular, we introduce partitioning of the observed set Ω, so that we use different partitions of Ω in each iteration. See Algorithm 2 for a pseudo-code of our variant of the alternating minimization approach.

Our use of some technical lemmas from [14] renders all the constants dependent on $\frac{n}{m}$. In what follows, a constant by default is assumed to depend on $\frac{n}{m}$. We believe that our results hold even with out this assumption but leave the extension to such case as a subject for future research. We now present our main result for (LRMC):

Algorithm 2 AltMinComplete: Alternating minimization for matrix completion

1: Input: observed set Ω, values $P_\Omega(M)$
2: Partition Ω into $2T+1$ subsets $\Omega_0, \cdots, \Omega_{2T}$ with each element of Ω belonging to one of the Ω_t with equal probability (sampling with replacement)
3: $\widehat{U}^0 = SVD(\frac{1}{p}P_{\Omega_0}(M), k)$ i.e., top-k left singular vectors of $\frac{1}{p}P_{\Omega_0}(M)$
4: Clipping step : Set all elements of $\widehat{U}^0$ that have magnitude greater than $\frac{2\mu\sqrt{k}}{\sqrt{n}}$ to zero and orthonormalize the columns of $\widehat{U}^0$
5: **for** $t = 0, \cdots, T-1$ **do**
6: $\quad \widehat{V}^{t+1} \leftarrow \text{argmin}_{V \in \mathbb{R}^{n\times k}} \ \|P_{\Omega_{t+1}}(\widehat{U}^t V^\dagger - M)\|_F^2$
7: $\quad \widehat{U}^{t+1} \leftarrow \text{argmin}_{U \in \mathbb{R}^{m\times k}} \ \|P_{\Omega_{T+t+1}}(U\left(\widehat{V}^{t+1}\right)^\dagger - M)\|_F^2$
8: **end for**
9: Return $X = \widehat{U}^T(\widehat{V}^T)^\dagger$

THEOREM 2.5. *Let $M = U^*\Sigma^*V^{*\dagger} \in \mathbb{R}^{m\times n}$ $(n \geq m)$ be a rank-k incoherent matrix, i.e., both U^* and V^* are μ-incoherent (see Definition 2.4). Also, let each entry of M be observed uniformly and independently with probability,*

$$p > C\frac{\left(\frac{\sigma_1^*}{\sigma_k^*}\right)^4 \mu^4 k^7 \log n \log\frac{k\|M\|_F}{\epsilon}}{m\delta_{2k}^2},$$

where $\delta_{2k} \leq \frac{\sigma_k^*}{C\sigma_1^*}$ and $C > 0$ is a global constant. Then w.h.p. for $T = C' \log\frac{\|M\|_F}{\epsilon}$, the outputs $\widehat{U}^T$ and V^T of Algorithm 2, with input $(\Omega, P_\Omega(M))$ (see Equation (2)) satisfy: $\left\|M - \widehat{U}^T\left(V^T\right)^\dagger\right\|_F \leq \epsilon$.

The above theorem implies that by observing $|\Omega| = O\left((\frac{\sigma_1^*}{\sigma_k^*})^6 k^7 n \log n \log(k\|M\|_F/\epsilon)\right)$ random entries of an incoherent M, AltMinComplete can recover M in $O(\log(1/\epsilon))$ steps. In terms of sample complexity ($|\Omega|$), our results show alternating minimization may require a bigger Ω than convex optimization, as our result has $|\Omega|$ depend on the condition number, required accuracy (ϵ) and worse dependence on k than known bounds. In contrast, trace-norm minimization based methods require $O(kn\log n)$ samples only.

Empirically however, this is not seen to be the case [10] and we leave further tightening of the sample complexity bounds for matrix completion as an open problem.

In terms of time complexity, we show that AltMinComplete needs time $O(|\Omega|k^2\log(1/\epsilon))$. This is in contrast to popular trace-norm minimization based methods that need $O(1/\sqrt{\epsilon})$ steps [1] and total time complexity of $O(|\Omega|n/\sqrt{\epsilon})$; note that the latter can potentially be quadratic in n. Furthermore, each step of such methods requires computation of the SVD of an $m \times n$ matrix. As mentioned earlier, interior point methods for trace-norm minimization also converge in $O(\log(1/\epsilon))$ steps but each iteration requires $O(n^5)$ steps and need storage of the entire $m \times n$ matrix X.

3. RELATED WORK

Alternating Minimization: Alternating minimization and its variants have been applied to several low-rank matrix estimation problems. For example, clustering [16], sparse PCA [25], non-negative matrix factorization [15], signed network prediction [9] etc. There are three main reasons for such wide applicability of this approach: a) low-memory footprint and fast iterations, b) flexible modeling, c) amenable to parallelization. However, despite such empirical success, this approach has largely been used as a heuristic and has had no theoretical analysis other than the guarantees of convergence to the *local minima* [23]. Ours is the first analysis of this approach for two practically important problems: a) matrix completion, b) matrix sensing.

After this paper was submitted, we became aware of [13] which provides an analysis of alternating minimization for matrix completion. Along with [13], ours is the first analysis of this approach for the problem of matrix completion. Moreover, ours is the first analysis of this approach for the problem of matrix sensing. .

Matrix Completion: This is the problem of completing a low-rank matrix from a few sampled entries. Candes and Recht [3] provided the first results on this problem, showing that under the random sampling and incoherence conditions (detailed above), $O(kn^{1.2}\log n)$ samples allow for recovery via convex trace-norm minimization; this was improved to $O(kn\log n)$ in [5]. For large matrices, this approach is not very attractive due to the need to store and update the entire matrix, and because iterative methods for trace norm minimization require $O(\frac{1}{\sqrt{\epsilon}})$ steps to achieve additive error of ϵ. Moreover, each such step needs to compute an SVD.

Another approach, in [14], involved taking a single SVD, followed by gradient descent on a Grassmanian manifold. However, *(a)* this is more expensive than alternating minimization as it needs to compute gradient over Grassmanian manifold which in general is a computationally intensive step, and *(b)* the analysis of the algorithm only guarantees asymptotic convergence, and in the worst case might take exponential time in the problem size.

The most closely related work to ours is [13], which provides guarantees for alternating minimization for the case of matrix completion. [13] shows that consistent recovery is possible if the sampling probability p scales as $\Omega\left(k\left(\frac{\sigma_1^*}{\sigma_k^*}\right)^8\frac{\log n}{m}\right)$. Our result is worse than theirs in the dependence on k while being better in the dependence on the condition number.

Recently, several other matrix completion type of problems have been studied in the literature. For example, robust PCA [6, 2], spectral clustering [12] etc. Here again, under additional assumptions, convex relaxation based methods have rigorous analysis but alternating minimization based algorithms continue to be algorithms of choice in practice.

Matrix Sensing: The general problem of matrix sensing was first proposed by [22]. They established recovery via trace norm minimization, assuming the sensing operator satisfies "restricted isometry" conditions. Subsequently, several other methods [10, 19] were proposed for this problem that also recovers the underlying matrix with optimal number of measurements and can give an ϵ-additive approximation in time $O(\log(1/\epsilon))$. But, similar to matrix completion, most of these methods require computing SVD of a large matrix at each step and hence have poor scalability to large problems.

We show that AltMinSense and AltMin-Completion provide more scalable algorithms for their respective problems. We demonstrate that these algorithms have geometric convergence to the optima, while each iteration is relatively cheap. For this, we assume conditions similar to those required by existing algorithms; albeit, with one drawback:

number of samples required by our analysis depend on the condition number of the underlying matrix M. For the matrix sensing problem, we remove this requirement by using a stagewise algorithm; we leave similar analysis for matrix completion as an open problem.

4. MATRIX SENSING

In this section, we study the matrix sensing problem (LRMS) and prove that if the measurement operator, $\mathcal{A}$, satisfies RIP then AltMinSense (Algorithm 1) recovers the underlying low-rank matrix *exactly* (see Theorem 2.2).

At a high level, we prove Theorem 2.2 by showing that the "distance" between subspaces spanned by $\widehat{V}^t$ (iterate at time t) and V^* decreases exponentially with t. This done based on the observation that once the (standard) matrix RIP condition (Definition 2.1) holds, alternating minimization can be viewed, and analyzed, as a **perturbed version of the power method.** This is easiest to see for the rank-1 case below; we detail this proof, and then the more general rank-k case.

In this paper, we use the following definition of distance between subspaces:

DEFINITION 4.1. *[8] Given two matrices $\widehat{U}, \widehat{W} \in \mathbb{R}^{m\times k}$, the (principal angle) distance between the subspaces spanned by the columns of $\widehat{U}$ and $\widehat{W}$ is given by:*

$$\text{dist}\left(\widehat{U},\widehat{W}\right) \stackrel{def}{=} \left\|U_\perp^\dagger W\right\|_2 = \left\|W_\perp^\dagger U\right\|_2$$

where U and W are orthonormal bases of the spaces Span $\left(\widehat{U}\right)$ and Span $\left(\widehat{W}\right)$, respectively. Similarly, $U_\perp$ and $W_\perp$ are any orthonormal bases of the perpendicular spaces Span $(U)^\perp$ and Span $(W)^\perp$, respectively.

Note: *(a)* The distance depends only on the spaces spanned by the columns of $\widehat{U}, \widehat{W}$, *(b)* if the ranks of $\widehat{U}$ and $\widehat{W}$ (i.e. the dimensions of their spans) are not equal, then dist $\left(\widehat{U},\widehat{W}\right) = 1$, and *(c)* dist $\left(\widehat{U},\widehat{W}\right) = 0$ if and only if they span the same subspace of $\mathbb{R}^m$.

We now present a theorem that bounds the distance between the subspaces spanned by $\widehat{V}^t$ and V^* and show that it decreases exponentially with t.

THEOREM 4.2. *Let $b = \mathcal{A}(M)$ where M and $\mathcal{A}$ satisfy assumptions given in Theorem 2.2. Then, the $(t+1)$-th iterates $\widehat{U}^{t+1}$, $\widehat{V}^{t+1}$ of AltMinSense satisfy:*

$$\text{dist}\left(\widehat{V}^{t+1},V^*\right) \le \frac{1}{4}\cdot\text{dist}\left(\widehat{U}^t,U^*\right) \ ,$$
$$\text{dist}\left(\widehat{U}^{t+1},U^*\right) \le \frac{1}{4}\cdot\text{dist}\left(\widehat{V}^{t+1},V^*\right)$$

where dist (U,W) *denotes the principal angle based distance (see Definition 4.1).*

Using Theorem 4.2, we are now ready to prove Theorem 2.2.

PROOF OF THEOREM 2.2. Assuming correctness of Theorem 4.2, Theorem 2.2 follows by using the following set of inequalities:

$$\begin{aligned}
\|M-\widehat{U}^T(\widehat{V}^T)^\dagger\|_F^2 &\stackrel{\zeta_1}{\le} \frac{1}{1-\delta_{2k}}\|\mathcal{A}(M-\widehat{U}^T(\widehat{V}^T)^\dagger)\|_2^2,\\
&\stackrel{\zeta_2}{\le} \frac{1}{1-\delta_{2k}}\|\mathcal{A}(M(I-V^T(V^T)^\dagger))\|_2^2,\\
&\stackrel{\zeta_3}{\le} \frac{1+\delta_{2k}}{1-\delta_{2k}}\|U^*\Sigma^*(V^*)^\dagger(I-V^T(V^T)^\dagger))\|_F^2,\\
&\stackrel{\zeta_4}{\le} \frac{1+\delta_{2k}}{1-\delta_{2k}}\|M\|_F^2\text{dist}^2\left(V^T,V^*\right) \stackrel{\zeta_5}{\le} \epsilon,
\end{aligned}$$

where V^T is an orthonormal basis of $\widehat{V}^T$, ζ_1 and ζ_3 follow by RIP, ζ_2 holds as $\widehat{U}^T$ is the least squares solution, ζ_4 follows from the definition of dist$(\cdot,\cdot)$ and finally ζ_5 follows from Theorem 4.2 and by setting T appropriately. □

To complete the proof of Theorem 2.2, we now need to prove Theorem 4.2. In the next section, we illustrate the main ideas of the proof of Theorem 4.2 by applying it to a rank-1 matrix i.e., when $k=1$. We then provide a proof of Theorem 4.2 for arbitrary k in Section 4.2.

4.1 Rank-1 Case

In this section, we provide a proof of Theorem 4.2 for the special case of $k=1$. That is, let $M = u^*\sigma^*(v^*)^\dagger$ s.t. $u^*\in\mathbb{R}^m$, $\|u^*\|_2=1$ and $v^*\in\mathbb{R}^n$, $\|v^*\|_2=1$. Also note that when $\widehat{u}$ and $\widehat{w}$ are vectors, dist$(\widehat{u},\widehat{w}) = 1-(u^\dagger w)^2$, where $u=\widehat{u}/\|\widehat{u}\|_2$ and $w=\widehat{w}/\|\widehat{w}\|_2$.

Consider the t-th update step in the AltMinSense procedure. As $\widehat{v}^{t+1} = \text{argmin}_{\widehat{v}}\sum_{i=1}^d\left(\widehat{u}^{t\dagger}A_i^\dagger\widehat{v}-\sigma^*u^{*\dagger}A_i^\dagger v^*\right)^2$, setting the gradient of the above objective function to 0, we obtain:

$$\left(\sum_{i=1}^d A_i u^t(u^t)^\dagger A_i^\dagger\right)\|\widehat{u}^t\|_2\widehat{v}^{t+1} = \sigma^*\left(\sum_{i=1}^d A_i u^t u^{*\dagger}A_i^\dagger\right)v^*,$$

where $u^t=\widehat{u}^t/\|\widehat{u}^t\|_2$. Now, let $B=\sum_{i=1}^d A_i u^t(u^t)^\dagger A_i^\dagger$ and $C=\sum_{i=1}^d A_i u^t(u^*)^\dagger A_i^\dagger$. Then,

$$\begin{aligned}
\|\widehat{u}^t\|_2\widehat{v}^{t+1} &= \sigma^*B^{-1}Cv^*,\\
&= \underbrace{\langle u^*,u^t\rangle\sigma^*v^*}_{\text{Power Method}} - \underbrace{B^{-1}\left(\langle u^*,u^t\rangle B-C\right)\sigma^*v^*}_{\text{Error Term}}. \quad (4)
\end{aligned}$$

Note that the first term in the above expression is the power method iterate (i.e., $M^\dagger u^t$). The second term is an error term and the goal is to show that it becomes smaller as u^t gets closer to u^*. Note that when $u^t=u^*$, the error term is 0 *irrespective* of the measurement operator $\mathcal{A}$.

Below, we provide a precise bound on the error term:

LEMMA 4.3. *Consider the error term defined in (4) and let $\mathcal{A}$ satisfy 2-RIP with constant δ_2. Then,*

$$\|B^{-1}\left(\langle u^*,u^t\rangle B-C\right)v^*\| \le \frac{3\delta_2}{1-3\delta_2}\sqrt{1-\langle u^t,u^*\rangle^2}$$

Using the above lemma, we now finish the proof of Theorem 4.2:

PROOF OF RANK-1 CASE OF THEOREM 4.2. Let $v^{t+1} =$

$\widehat{v}^{t+1}/\|\widehat{v}^{t+1}\|_2$. Now, using (4) and Lemma 4.3:,

$$\langle v^{t+1}, v^*\rangle = \frac{\langle \widehat{v}^{t+1}, v^*\rangle}{\|\widehat{v}^{t+1}\|} = \frac{\langle \widehat{v}^{t+1}/\sigma^*, v^*\rangle}{\|\widehat{v}^{t+1}/\sigma^*\|}$$
$$\leq \frac{\langle u^*, u^t\rangle - \widehat{\delta}_2\sqrt{1-\langle u^*, u^t\rangle^2}}{\sqrt{\left(\langle u^*, u^t\rangle - \widehat{\delta}_2\sqrt{1-\langle u^*, u^t\rangle^2}\right)^2 + \widehat{\delta}_2^2\left(1-\langle u^*, u^t\rangle^2\right)}},$$

where $\widehat{\delta}_2 = \frac{3\delta_2}{1-3\delta_2}$. That is,

$$\mathrm{dist}^2(v^{t+1}, v^*) \leq \frac{\widehat{\delta}_2^2(1-\langle u^*, u^t\rangle^2)}{(\langle u^*, u^t\rangle - \widehat{\delta}_2\sqrt{1-\langle u^*, u^t\rangle^2})^2 + \widehat{\delta}_2^2(1-\langle u^*, u^t\rangle^2)},$$

Hence, *assuming* $\langle u^*, u^t\rangle \geq 5\widehat{\delta}_2$, $\mathrm{dist}(v^{t+1}, v^*) \leq \frac{1}{4}\mathrm{dist}(u^t, u^*)$. As $\mathrm{dist}(u^{t+1}, u^*)$ and $\mathrm{dist}(v^{t+1}, v^*)$ are decreasing with t (from the above bound), we only need to show that $\langle u^0, u^t\rangle \geq 5\widehat{\delta}_2$. Recall that $\widehat{u}^0$ is obtained by using one step of the Singular Value Projection (SVP) algorithm [10]. Hence, using Lemma 2.1 of [10]:

$$\|\sigma_1^*(I - u^0(u^0)^\dagger)u^*)\|_2^2 \leq \|M - \widehat{u}^0(\widehat{v}^0)^\dagger\|_F^2 \leq 2\delta_2\|M\|_F^2.$$

Therefore, $\langle u^0, u^*\rangle \geq \sqrt{1-2\delta_2} \geq 5\widehat{\delta}_2$ assuming $\delta_2 \leq \frac{1}{100}$. □

4.2 Rank-k Case

In this section, we present the proof of Theorem 4.2 for arbitrary k, i.e., when M is a rank-k matrix (with SVD $U^*\Sigma^*(V^*)^\dagger$).

Similar to the analysis for the rank-1 case (Section 4.1), we show that even for arbitrary k, the updates of AltMinSense are essentially power-method type updates but with a bounded error term whose magnitude decreases with each iteration.

However, directly analyzing iterates of AltMinSense is a bit tedious due to non-orthonormality of intermediate iterates $\widehat{U}$. Instead, **for analysis only** we consider the iterates of a modified version of AltMinSense, where we explicitly orthonormalize each iterate using the QR-decomposition[3]. In particular, suppose we replace steps 4 and 5 of AltMinSensewith the following

$$\begin{aligned}
\widehat{U}^t &= U^t R_U^t \quad \text{(QR decomposition)},\\
\widehat{V}^{t+1} &\leftarrow \operatorname*{argmin}_V \|\mathcal{A}(U^tV^\dagger) - b\|_2^2,\\
\widehat{V}^{t+1} &= V^{t+1}R_V^{t+1} \quad \text{(QR decomposition)}\\
\widehat{U}^{t+1} &\leftarrow \operatorname*{argmin}_U \|\mathcal{A}(U(V^{t+1})^\dagger) - b\|_2^2
\end{aligned} \tag{5}$$

In our algorithm, in each iterate both $\widehat{U}^t, \widehat{V}^t$ remain full-rank because $\mathrm{dist}\left(U^t, U^*\right) < 1$; with this, the following lemma implies that the spaces spanned by the iterates in our AltMinSense algorithm are *exactly the same* as the respective ones by the iterates of the above modified version (and hence the distances $\mathrm{dist}(\widehat{U}^t, U^*)$ and $\mathrm{dist}(\widehat{V}^t, V^*)$ are also the same for the two algorithms).

[3]The QR decomposition factorizes a matrix into an orthonormal matrix (a basis of its column space) and an upper triangular matrix; that is given $\widehat{S}$ it computes $\widehat{S} = SR$ where S has orthonormal columns and R is upper triangular. If $\widehat{S}$ is full-rank, so are S and R.

LEMMA 4.4. *Let $\widehat{U}^t$ be the t^{th} iterate of our AltMinSense algorithm, and $\widetilde{U}^t$ of the modified version stated above. Suppose also that both $\widehat{U}^t, \widetilde{U}^t$ are full-rank, and span the same subspace. Then the same will be true for the subsequent iterates for the two algorithms, i.e. $Span(\widehat{V}^{t+1}) = Span(\widetilde{V}^{t+1})$, $Span(\widehat{U}^{t+1}) = Span(\widetilde{U}^{t+1})$, and all matrices at iterate $t+1$ will be full-rank.*

In light of this, we will now prove Theorem 4.2 **with** the new QR-based iterates (5).

LEMMA 4.5. *Let $\widehat{U}^t$ be the t-th step iterate of AltMinSense and let $U^t, \widehat{V}^{t+1}$ and V^{t+1} be obtained by Update (5). Then,*

$$\widehat{V}^{t+1} = \underbrace{V^*\Sigma^*U^{*\dagger}U^t}_{\substack{\text{Power-method}\\ \text{Update}}} - \underbrace{F}_{\substack{\text{Error}\\ \text{Term}}}, \quad V^{t+1} = \widehat{V}^{t+1}(R^{(t+1)})^{-1}, \tag{6}$$

where F is an error matrix defined in (8) and $R^{(t+1)}$ is a triangular matrix obtained using QR-decomposition of $\widehat{V}^{t+1}$.

Before we give an expression for the error matrix F, we define the following notation. Let $v^* \in \mathbb{R}^{nk}$ be given by: $v^* = vec(V^*)$, i.e., $v^* = \left[v_1^{*\dagger}v_2^{*\dagger}\dots v_k^{*\dagger}\right]^\dagger$. Define B, C, D, S as follows:

$$B \stackrel{\text{def}}{=} \begin{bmatrix} B_{11} & \cdots & B_{1k}\\ \vdots & \ddots & \vdots\\ B_{k1} & \cdots & B_{kk}\end{bmatrix}, C \stackrel{\text{def}}{=} \begin{bmatrix} C_{11} & \cdots & C_{1k}\\ \vdots & \ddots & \vdots\\ C_{k1} & \cdots & C_{kk}\end{bmatrix},$$
$$D \stackrel{\text{def}}{=} \begin{bmatrix} D_{11} & \cdots & D_{1k}\\ \vdots & \ddots & \vdots\\ D_{k1} & \cdots & D_{kk}\end{bmatrix}, S \stackrel{\text{def}}{=} \begin{bmatrix} \sigma_1^*I_n & \dots & 0_n\\ \vdots & \ddots & \vdots\\ 0_n & \dots & \sigma_k^*I_n\end{bmatrix}. \tag{7}$$

where , for $1 \leq p, q \leq k$: $B_{pq} \stackrel{\text{def}}{=} \sum_{i=1}^d A_i u_p^t {u_q^t}^\dagger A_i^\dagger$, $C_{pq} \stackrel{\text{def}}{=} \sum_{i=1}^d A_i u_p^t u_q^{*\dagger} A_i^\dagger$, and, $D_{pq} \stackrel{\text{def}}{=} \langle u_p^t, u_q^*\rangle \mathbb{I}_{n\times n}$. Recall that, u_p^t is the p-th column of U^t and u_q^* is the q-th left singular vector of the underlying matrix $M = U^*\Sigma^*(V^*)^\dagger$. Finally F is obtained by "de-stacking" the vector $B^{-1}(BD - C)Sv^*$ i.e., the i^{th} column of F is given by:

$$F_i \stackrel{\text{def}}{=} \begin{bmatrix} \left(B^{-1}(BD-C)Sv^*\right)_{ni+1}\\ \left(B^{-1}(BD-C)Sv^*\right)_{ni+2}\\ \vdots\\ \left(B^{-1}(BD-C)Sv^*\right)_{ni+n}\end{bmatrix}, F \stackrel{\text{def}}{=} [F_1\ F_2\ \cdots\ F_k]. \tag{8}$$

Note that the notation above should have been B^t, C^t and so on. We suppress the dependence on t for notational simplicity. Now, from Update (6), we have

$$V^{t+1} = \widehat{V}^{t+1}{R^{(t+1)}}^{-1} = \left(V^*\Sigma^*U^{*\dagger}U^t - F\right){R^{(t+1)}}^{-1}$$
$$\Rightarrow V_\perp^{*\dagger}V^{t+1} = -V_\perp^{*\dagger}F{R^{(t+1)}}^{-1}. \tag{9}$$

where $V_\perp^*$ is an orthonormal basis of $\mathrm{Span}\left(v_1^*, v_2^*, \cdots, v_k^*\right)^\perp$. Therefore,

$$dist(V^*, V^{t+1}) = \|V_\perp^{*\dagger}V^{t+1}\|_2 = \|V_\perp^{*\dagger}F{R^{(t+1)}}^{-1}\|_2$$
$$\leq \|F(\Sigma^*)^{-1}\|_2\|\Sigma^*{R^{(t+1)}}^{-1}\|_2.$$

Now, we break down the proof of Theorem 4.2 into the following two steps:

- show that $\|F\|_2$ is small (Lemma 4.6) and
- show that $\|R^{(t+1)^{-1}}\|_2$ is small(Lemma 4.7).

We will now state the two corresponding lemmas. The first lemma bounds the spectral norm of $F(\Sigma^*)^{-1}$.

LEMMA 4.6. *Let linear measurement $\mathcal{A}$ satisfy RIP for all $2k$-rank matrices and let $b = \mathcal{A}(M)$ with $M \in \mathbb{R}^{m\times n}$ being a rank-k matrix. Then, spectral norm of error matrix F (see Equation 6) after t-th iteration update satisfy:*

$$\left\|F(\Sigma^*)^{-1}\right\|_2 \leq \frac{\delta_{2k}k\sigma_1^*}{1-\delta_{2k}} dist(U^t, U^*). \tag{10}$$

The following lemma bounds the spectral norm of $R^{(t+1)^{-1}}$.

LEMMA 4.7. *Let linear measurement $\mathcal{A}$ satisfy RIP for all $2k$-rank matrices and let $b = \mathcal{A}(M)$ with $M \in \mathbb{R}^{m\times n}$ being a rank-k matrix. Then,*

$$\|\Sigma^*(R^{(t+1)})^{-1}\|_2 \leq \frac{\sigma_1^*/\sigma_k^*}{\sqrt{1-\text{dist}^2\left(U^t,U^*\right)} - \frac{(1/\sigma_k^*)\delta_{2k}k\text{dist}(U^t,U^*)}{1-\delta_{2k}}}. \tag{11}$$

With the above two lemmas, we now prove Theorem 4.2.

PROOF OF THEOREM 4.2. Using (9), (10) and (11), we obtain the following:

$$\begin{aligned}
\text{dist}\left(V^{t+1},V^*\right) &= \left\|V_\perp^{*\dagger}V^{t+1}\right\|_2, \\
&\leq \left\|V_\perp^{*\dagger}FR^{(t+1)^{-1}}\right\|_2, \\
&\leq \|V_\perp^*\|_2\|F\|_2\left\|R^{(t+1)^{-1}}\right\|_2 \\
&\leq \frac{(1/\sigma_k^*)\delta_{2k}k\cdot\text{dist}\left(U^t,U^*\right)}{(1-\delta_{2k})L},
\end{aligned} \tag{12}$$

where $L = \sqrt{1-\text{dist}\left(U^t,U^*\right)^2} - \frac{(\sigma_1^*/\sigma_k^*)\delta_{2k}k\text{dist}\left(U^t,U^*\right)}{1-\delta_{2k}}$. Also, note that U^0 is obtained using SVD of $\sum_i A_i b_i$.

$$\begin{aligned}
&\|\mathcal{A}(U^0\Sigma^0V^0 - U^*\Sigma^*(V^*)^\dagger\|_2^2 \leq 4\delta_{2k}\|\mathcal{A}(U^*\Sigma^*(V^*)^\dagger)\|_2^2, \\
\Rightarrow&\|U^0\Sigma^0V^0 - U^*\Sigma^*(V^*)^\dagger\|_F^2 \leq 4\delta_{2k}(1+3\delta_{2k})\|\Sigma^*\|_F^2, \\
\Rightarrow&\|U^0(U^0)^\dagger U^*\Sigma^*(V^*)^\dagger - U^*\Sigma^*(V^*)^\dagger\|_F^2 \leq 6\delta_{2k}\|\Sigma^*\|_F^2, \\
\Rightarrow&(\sigma_k^*)^2\|(U^0(U^0)^\dagger - I)U^*\|_F^2 \leq 6\delta_{2k}k(\sigma_1^*)^2, \\
\Rightarrow&\text{dist}(U^0,U^*) \leq \sqrt{6\delta_{2k}k}\left(\frac{\sigma_1^*}{\sigma_k^*}\right) < \frac{1}{2}.
\end{aligned} \tag{13}$$

Using (12) with $\text{dist}\left(U^0,U^*\right) < \frac{1}{2}$ and $\delta_{2k} < \frac{1}{24(\sigma_1^*/\sigma_k^*)^2k}$, we obtain: $\text{dist}\left(V^t,V^*\right) < \frac{1}{4}\text{dist}\left(U^t,U^*\right)$. Similarly we can show that $\text{dist}\left(U^{t+1},U^*\right) < \frac{1}{4}\text{dist}\left(V^t,V^*\right)$. □

5. MATRIX COMPLETION

In this section, we study the Matrix Completion problem (LRMC) and show that, assuming k and $\frac{\sigma_1^*}{\sigma_k^*}$ are constant, AltMinComplete (Algorithm 2) recovers the underlying matrix M using only $O(n\log n)$ measurements (i.e., we prove Theorem 2.5).

As mentioned, while observing elements in Ω constitutes a linear map, matrix completion is different from matrix sensing because the map does not satisfy RIP. The (now standard) approach is to assume incoherence of the true matrix M, as done in Definition 2.4. With this, and the random sampling of Ω, matrix completion exhibits similarities to matrix sensing. For our analysis, we can again use the fact that incoherence allows us to view alternating minimization as a perturbed power method, whose error we can control.

However, there are important differences between the two problems, which make the analysis of completion more complicated. Chief among them is the fact that we need to establish *the incoherence of each iterate*. For the first initialization $\widehat{U}^0$, this necessitates the "clipping" procedure (described in step 4 of the algorithm). For the subsequent steps, this requires the partitioning of the observed Ω into $2T+1$ sets (as described in step 2 of the algorithm).

As in the case of matrix sensing, we prove our main result for matrix completion (Theorem 2.5) by first establishing a geometric decay of the distance between the subspaces spanned by $\widehat{U}^t, \widehat{V}^t$ and U^*, V^* respectively.

THEOREM 5.1. *Under the assumptions of Theorem 2.5, the $(t+1)^{th}$ iterates $\widehat{U}^{t+1}$ and $\widehat{V}^{t+1}$ satisfy the following property w.h.p.:*

$$\begin{aligned}
&\text{dist}\left(\widehat{V}^{t+1},V^*\right) \leq \frac{1}{4}\text{dist}\left(\widehat{U}^t,U^*\right) \text{ and} \\
&\text{dist}\left(\widehat{U}^{t+1},U^*\right) \leq \frac{1}{4}\text{dist}\left(\widehat{V}^{t+1},V^*\right), \quad \forall\ 1 \leq t \leq T.
\end{aligned}$$

We use the above result along with incoherence of M to prove Theorem 2.5.

Now, similar to the matrix sensing case, alternating minimization needs an initial iterate that is close enough to U^* and V^*, from where it will then converge. To this end, Steps $3-4$ of Algorithm 2 use SVD of $P_\Omega(M)$ followed by clipping to initialize $\widehat{U}^0$. While the SVD step guarantees that $\widehat{U}^0$ is close enough to U^*, it might not remain incoherent. To maintain incoherence, we introduce an extra clipping step which guarantees incoherence of $\widehat{U}^0$ while also ensuring that $\widehat{U}^0$ is close enough to U^* (see Lemma 5.2)

LEMMA 5.2. *Let M, Ω, p be as defined in Theorem 2.5. Also, let U^0 be the initial iterate obtained by step 4 of Algorithm 2. Then, w.h.p. we have*

- $\text{dist}\left(U^0,U^*\right) \leq \frac{1}{2}$ *and*
- U^0 *is incoherent with parameter* $4\mu\sqrt{k}$.

The above lemma guarantees a "good" starting point for alternating minimization. Using this, we now present a proof of Theorem 5.1. Similar to the sensing section, we first explain key ideas of our proof using rank-1 example. Then in Section 5.2 we extend our proof to general rank-k matrices.

5.1 Rank-1 Case

Consider the rank-1 matrix completion problem where $M = \sigma^* u^*(v^*)^\dagger$. Now, the t-th step iterates $\widehat{v}^{t+1}$ of Algorithm 2 are given by:

$$\widehat{v}^{t+1} = \underset{\widehat{v}}{\text{argmin}} \sum_{(i,j)\in\Omega} (M_{ij} - \widehat{u}_i^t\widehat{v}_j)^2.$$

Let $u^t = \widehat{u}^t/\|\widehat{u}^t\|_2$. Then, $\forall j$:

$$\|\widehat{u}^t\|_2 \sum_{i:(i,j)\in\Omega} (u_i^t)^2 \widehat{v}_j^{t+1} = \sigma^* \sum_{i:(i,j)\in\Omega} u_i^t u_i^* v_j^*$$

$$\Rightarrow \|\widehat{u}^t\|_2 \widehat{v}_j^{t+1} = \frac{\sigma^*}{\sum_{i:(i,j)\in\Omega}(u_i^t)^2} \sum_{i:(i,j)\in\Omega} u_i^t u_i^* v_j^*$$

$$= \sigma^* \langle u^t, u^*\rangle v_j^* - \frac{\sigma^*(\langle u^t, u^*\rangle \sum_{i:(i,j)\in\Omega}(u_i^t)^2 v_j^* - \sum_{i:(i,j)\in\Omega} u_i^t u_i^* v_j^*)}{\sum_{i:(i,j)\in\Omega}(u_i^t)^2}. \tag{14}$$

Hence,

$$\|\widehat{u}^t\|_2 \widehat{v}^{t+1} = \underbrace{\langle u^*, u^t\rangle \sigma^* v^*}_{\text{Power Method}} - \underbrace{\sigma^* B^{-1}\left(\langle u^t, u^*\rangle B - C\right) v^*}_{\text{Error Term}}, \tag{15}$$

where $B, C \in \mathbb{R}^{n\times n}$ are diagonal matrices, such that,

$$B_{jj} = \frac{\sum_{i:(i,j)\in\Omega}(u_i^t)^2}{p}, \quad C_{jj} = \frac{\sum_{i:(i,j)\in\Omega} u_i^t u_i^*}{p}. \tag{16}$$

Note the similarities between the update (15) and the rank-1 update (4) for the sensing case. Here again, it is essentially a power-method update (first term) along with a bounded error term (see Lemma 5.3). Using this insight, we now prove Theorem 5.1 for the special case of rank-1 matrices. Our proof can be divided in three major steps:

- *Base Case*: Show that $u^0 = \widehat{u}^0/\|\widehat{u}^0\|_2$ is incoherent and have small distance to u^* (see Lemma 5.2).
- *Induction Step (distance)*: Assuming $u^t = \widehat{u}^t/\|\widehat{u}^t\|_2$ to be incoherent and that u^t has a small distance to u^*, v^{t+1} decreases distances to v^* by at least a constant factor.
- *Induction Step (incoherence)*: Show incoherence of v^{t+1}, while assuming incoherence of u^t (see Lemma 5.4)

We first prove the second step of our proof. To this end, we provide the following lemma that bounds the error term.

LEMMA 5.3. *Let M, p, Ω, u^t be as defined in Theorem 2.5. Also, let u^t be a unit vector with incoherence parameter $\mu_1 = \frac{6(1+\delta_2)\mu}{1-\delta_2}$. Then, w.p. at least $1 - \frac{1}{n^3}$:*

$$\|B^{-1}\left(\langle u^*, u^t\rangle B - C\right) v^*\|_2 \le \frac{\delta_2}{1-\delta_2}\sqrt{1 - \langle u^t, u^*\rangle^2}.$$

Multiplying (15) with v^* and using Lemma 5.3, we get:

$$\|\widehat{u}^t\|_2 \langle \widehat{v}^{t+1}, v^*\rangle \ge \sigma^* \langle u^t, u^*\rangle - 2\sigma^*\delta_2\sqrt{1 - \langle u^t, u^*\rangle^2}, \tag{17}$$

where $\delta_2 < \frac{1}{12}$ is a constant defined in the Theorem statement and is similar to the RIP constant in Section 4.

Similarly, by multiplying (15) with $v_\perp$ (where $\langle v_\perp^*, v^*\rangle = 0$ and $\|v_\perp^*\|_2 = 1$) and using Lemma 5.3:

$$\|\widehat{u}^t\|_2 \langle \widehat{v}^{t+1}, v_\perp^*\rangle \le 2\sigma^*\delta_2\sqrt{1 - \langle u^t, u^*\rangle^2}.$$

Using the above two equations:

$$1 - \langle v^{t+1}, v^*\rangle^2 \le \frac{4\delta_2^2(1 - \langle u^t, u^*\rangle^2)}{(\langle u^t, u^*\rangle - 2\delta_2\sqrt{1 - \langle u^t, u^*\rangle^2})^2 + (2\delta_2\sqrt{1 - \langle u^t, u^*\rangle^2})^2}.$$

Assuming, $\langle v^{t+1}, v^*\rangle \ge 6\delta_2$,

$$\text{dist}(v^{t+1}, v^*) = \sqrt{1 - \langle v^{t+1}, v^*\rangle^2} \le \frac{1}{4}\sqrt{1 - \langle u^t, u^*\rangle^2}.$$

Using same arguments, we can show that, $\text{dist}(u^{t+1}, u^*) \le \text{dist}(v^{t+1}, v^*)/4$. Hence, after $O(\log(1/\epsilon))$ iterations, $\text{dist}(u^t, u^*) \le \epsilon$ and $\text{dist}(v^{t+1}, v^*) \le \epsilon$. This proves our second step.

We now provide the following lemma to prove the third step. We stress that v^{t+1} does not increase the incoherence parameter (μ_1) when compared to that of u^t.

LEMMA 5.4. *Let M, p, Ω be as defined in Theorem 2.5. Also, let u^t be a unit vector with incoherence parameter $\mu_1 = \frac{6(1+\delta_2)\mu}{1-\delta_2}$. Then, w.p. at least $1 - \frac{1}{n^3}$, v^{t+1} is also μ_1 incoherent.*

Finally, for the base case we need that u^0 is μ_1 incoherent and also $\langle u^0, u^*\rangle \ge 6\delta_2$. This follows directly by using Lemma 5.2 and the fact that $\delta_2 \le 1/12$.

Note that, to obtain an error of ϵ, AltMinComplete needs to run for $O\left(\log \frac{\|M\|_F}{\epsilon}\right)$ iterations. Also, we need to sample a fresh Ω at each iteration of AltMinComplete. Hence, the total number of samples needed by AltMinComplete is $O\left(\log \frac{\|M\|_F}{\epsilon}\right)$ larger than the number of samples required per step.

5.2 Rank-k case

We now extend our proof of Theorem 5.1 to matrices with arbitrary rank. Here again, we show that the AltMinComplete algorithm reduces to power method with bounded perturbation at each step.

Similar to the matrix sensing case, we analyze the following QR decomposition based update instead of directly analyzing the updates of Algorithm 2:

$$\begin{aligned}
\widehat{U}^t &= U^t R_U^t \quad \text{(QR decomposition)},\\
\widehat{V}^{t+1} &= \operatorname*{argmin}_{\widehat{V}} \|P_\Omega(U^t\widehat{V}^\dagger) - P_\Omega(M)\|_F^2,\\
\widehat{V}^{t+1} &= V^{t+1}R_V^{t+1}. \quad \text{(QR decomposition)},\\
\widehat{U}^{t+1} &= \operatorname*{argmin}_{\widehat{U}} \|P_\Omega(\widehat{U}(V^{t+1})^\dagger) - P_\Omega(M)\|_F^2.
\end{aligned} \tag{18}$$

Here again, we would stress that the updates output exactly the same matrices at the end of each iteration and we prefer QR-based updates due to notational ease.

Now, as matrix completion is a special case of matrix sensing, Lemma 4.5 characterizes the updates of the AltMinComplete algorithm (see Algorithm 2). That is,

$$\begin{aligned}
\widehat{V}^{t+1} &= \underbrace{V^*\Sigma^* U^{*\dagger} U^t}_{\text{Power-method Update}} - \underbrace{F}_{\text{Error Term}},\\
V^{t+1} &= \widehat{V}^{t+1}(R^{(t+1)})^{-1},
\end{aligned} \tag{19}$$

where F is the error matrix defined in (8) and $R^{(t+1)}$ is a upper-triangular matrix obtained using QR-decomposition of $\widehat{V}^{t+1}$. See (7) for the definition of B, C, D, and S.

Also, note that for the special case of matrix completion, $B_{pq}, C_{pq}, 1 \le p, q \le k$ are *diagonal matrices* with

$$(B_{pq})_{jj} = \frac{1}{p}\sum_{i:(i,j)\in\Omega} U_{ip}^t U_{iq}^t, \quad (C_{pq})_{jj} = \frac{1}{p}\sum_{i:(i,j)\in\Omega} U_{ip}^t U_{iq}^*.$$

We use this structure to further simplify the update equation. We first define matrices $B^j, C^j, D^j \in \mathbb{R}^{k\times k}, 1 \le i \le n$:

$$B^j = \frac{1}{p}\sum_{i:(i,j)\in\Omega} (U^t)^{(i)}(U^t)^{(i)\dagger}, \quad C^j = \frac{1}{p}\sum_{i:(i,j)\in\Omega} (U^t)^{(i)}(U^*)^{(i)\dagger},$$

and $D^j = (U^t)^\dagger U^*$. Using the above notation, (19) decouples into n equations of the form $(1 \le j \le n)$:

$$(V^{t+1})^{(j)} = (V^*)^{(j)}(D^j - (B^j)^{-1}(B^j D^j - C^j))(R^{(t+1)})^{-1}, \tag{20}$$

where $(V^{t+1})^{(j)}$ and $(V^*)^{(j)}$ denote the j^{th} rows of V^{t+1} and V^* respectively.

Using the above notation, we now provide a proof of Theorem 5.1 for the general rank-k case.

PROOF OF THEOREM 5.1. Multiplying the update equation (19) on the left by $(V_\perp^*)^\dagger$, we get: $(V_\perp^*)^\dagger \widehat{V}^{t+1} = -(V_\perp^*)^\dagger F(R^{(t+1)})^{-1}$. That is,

$$\begin{aligned}\text{dist}(V^*, V^{t+1}) &= \|V_\perp^{*\dagger} V^{(t+1)}\|_2 = \|V_\perp^{*\dagger} F R^{(t+1)^{-1}}\|_2 \\ &\le \|F\|_2 \|R^{(t+1)^{-1}}\|_2.\end{aligned}$$

Now, similar to the sensing case (see Section 4.2) we break down our proof into the following two steps:

- Bound $\|F\|_2$ (Lemma 5.6) and
- Bound $\|R^{(t+1)^{-1}}\|_2$, i.e., the minimum singular value of $R^{(t+1)}$ (Lemma 5.7).

Using Lemma 5.6 and Lemma 5.7, w.p. at least $1 - 1/n^3$,

$$\begin{aligned}\text{dist}(V^*, V^{t+1}) &\le \|F\|_2 \|R^{(t+1)^{-1}}\|_2 \\ &\le \frac{\sigma_1^* \delta_{2k}/(1-\delta_{2k}) \cdot \text{dist}\left(U^{(t)}, U^*\right)}{\sigma_k^* \sqrt{1 - \text{dist}\left(U^{(t)}, U^*\right)^2} - \frac{\sigma_1^* \delta_{2k} \text{dist}\left(U^{(t)}, U^*\right)}{1-\delta_{2k}}}.\end{aligned}$$

Now, using Lemma 5.2 we get: $\text{dist}(U^t, U^*) \le \text{dist}(U^0, U^*) \le \frac{1}{2}$. By selecting $\delta_{2k} < \frac{\sigma_k^*}{C\sigma_1^*}$, i.e., $p \ge \frac{C(\sigma_1^*)^2 k^4 \log n}{m(\sigma_k^*)^2}$ and using above two inequalities:

$$\text{dist}(V^{t+1}, V^*) \le \frac{1}{4}\text{dist}(U^t, U^*).$$

Furthermore, using Lemma 5.5 we get that V^{t+1} is μ_1 incoherent. Hence, using similar arguments as above, we also get: $\text{dist}(U^{t+1}, U^*) \le \left(\frac{1}{4}\right)\text{dist}(V^{t+1}, V^*)$. □

We now provide lemmas required by our above given proof.

We first provide a lemma to bound incoherence of V^{t+1}, assuming incoherence of U^t.

LEMMA 5.5. *Let M, Ω, p be as defined in Theorem 2.5. Also, let U^t be the t-th step iterate obtained by (18). Let U^t be $\mu_1 = \frac{16\sigma_1^* \mu\sqrt{k}}{\sigma_k^*}$ incoherent. Then, w.p. at least $1 - 1/n^3$, iterate $V^{(t+1)}$ is also μ_1 incoherent.*

We now bound the error term (F) in AltMin update (19).

LEMMA 5.6. *Let F be the error matrix defined by (8) (also see (19)) and let U^t be a μ_1-incoherent orthonormal matrix obtained after $(t-1)^{th}$ update. Also, let M, Ω, and p satisfy assumptions of Theorem 2.5. Then, w.p. at least $1 - 1/n^3$:*

$$\|F\|_2 \le \frac{\delta_{2k}\sigma_1^*}{1-\delta_{2k}} dist(U^t, U^*).$$

Next, we present a lemma to bound $\|(R^{(t+1)})^{-1}\|_2$.

Algorithm 3 Stage-AltMin: Stagewise Alternating Minimization for Matrix Sensing

1: Input: $b, \mathcal{A}$
2: $\widehat{U}^T \leftarrow []$, $\widehat{V}^T \leftarrow []$
3: **for** $i = 1, \cdots, k$ **do**
4: $[\widehat{U}^0_{1:i}\ \widehat{V}^0_{1:i}]$ = top i-singular vectors of $\left(\widehat{U}^T_{1:i-1}(\widehat{V}^T_{1:i-1})^\dagger - \frac{3}{4}\mathcal{A}^T(\mathcal{A}(\widehat{U}^T_{1:i-1}(\widehat{V}^T_{1:i-1})^\dagger) - b)\right)$ i.e., one step of SVP [10]
5: **for** $t = 0, \cdots, T-1$ **do**
6: $\widehat{V}^{t+1}_{1:i} \leftarrow \text{argmin}_{V \in \mathbb{R}^{n\times i}} \|\mathcal{A}(\widehat{U}^t_{1:i} V^\dagger) - b\|_2^2$
7: $\widehat{U}^{t+1}_{1:i} \leftarrow \text{argmin}_{U \in \mathbb{R}^{m\times i}} \|\mathcal{A}(U_{1:i}(\widehat{V}^{t+1}_{1:i})^\dagger) - b\|_2^2$
8: **end for**
9: **end for**
10: Output: $X = \widehat{U}^T_{1:i}(\widehat{V}^T_{1:i})^\dagger$

LEMMA 5.7. *Let $R^{(t+1)}$ be the lower-triangular matrix obtained by QR decomposition of $\widehat{V}^{t+1}$ (see (19)) and let U^t be a μ_1-incoherent orthonormal matrix obtained after $(t-1)^{th}$ update. Also, let M and Ω satisfy assumptions of Theorem 2.5. Then,*

$$\|(R^{(t+1)})^{-1}\|_2 \le \frac{1/\sigma_k^*}{\sqrt{1 - \text{dist}^2\left(U^{(t)}, U^*\right)} - \frac{(\sigma_1^*/\sigma_k^*)\delta_{2k} k \text{dist}(U^{(t)}, U^*)}{1-\delta_{2k}}}$$

Lemma follows by exactly the same proof as that of Lemma 4.7 for the matrix sensing case.

6. STAGEWISE ALTMIN ALGORITHM

In Section 4, we showed that if $\delta_{2k} \le \frac{(\sigma_k^*)^2}{(\sigma_1^*)^2 k}$ then AltMinSense (Algorithm 1) recovers the underlying matrix. This means that, $d = \frac{(\sigma_1^*)^4}{(\sigma_k^*)^4} k^2 n \log n$ random Gaussian measurements (assume $m \le n$) are required to recover M. For matrices with large condition number (σ_1^*/σ_k^*), this would be significantly larger than the information theoretic bound of $O(kn \log n/k)$ measurements.

To alleviate this problem, we present a modified version of AltMinSense called Stage-AltMin. Stage-AltMin proceeds in k stages where in the i-th stage, a rank-i problem is solved. The goal of the i-th stage is to recover top i-singular vectors of M, up to $O(\sigma_{i+1}^*)$ error.

Specifically, we initialize the i-th stage of our algorithm using one step of the SVP algorithm [10] (see Step 4 of Algorithm 3). We then show that, if $\delta_{2k} \le \frac{1}{10k}$, then Stage-AltMin (Steps 6, 7 of Algorithm 3) decreases the error $\|M - \widehat{U}^T_{1:i}(\widehat{V}^T_{1:i})^\dagger\|_F$ to $O(\sigma_{i+1}^*)$. Hence, after k steps, the error decreases to $O(\sigma_{k+1}^*) = 0$. Note that, $\widehat{U}^t_{1:i} \in \mathbb{R}^{m\times i}$ represents the t-th step iterate (U) in the i-th stage; $\widehat{V}^t_{1:i} \in \mathbb{R}^{n\times i}$ is also defined similarly.

Recall that, the main problem with our analysis of AltMinSense is that if $\sigma_i \gg \sigma_{i+1}$ (for some i) then $\delta_{2k} \le \frac{(\sigma_{i+1}^*)^2}{(\sigma_i^*)^2 k}$ would need to be small. However, in such a scenario, the i-th stage of Algorithm 3 can be thought of as solving a noisy sensing problem where the goal is to recover $M_i \overset{\text{def}}{=} U^*_{1:i}\Sigma^*_{1:i}(V^*_{1:i})^\dagger$ using noisy measurements $b = \mathcal{A}(U^*_{1:i}\Sigma^*_{1:i}(V^*_{1:i})^\dagger + N)$ where noise matrix $N \overset{\text{def}}{=} U^*_{i+1:k}\Sigma^*_{i+1:k}(V^*_{i+1:k})^\dagger$. Here M_i and N represent the top i singular components and last $k-i$ singular components of M respectively. Hence, using noisy-case type analysis we show that the error $\|M - \widehat{U}^t(\widehat{V}^t)^\dagger\|_F$ decreases to $O(\sigma_{i+1}^*)$.

PROOF OUTLINE OF THEOREM 2.3. We prove the theorem using mathematical induction.

Base Case: After the 0-th step, error is: $\|M\|_F^2 \leq \sum_{j=1}^{k} \sigma_j^2 \leq k\sigma_1^2$. Hence, base case holds.

Induction Step: Here, assuming that the error bound holds for $(i-1)$-th stage, we prove the error bound for the i-th stage.

Our proof proceeds in two steps. First, we show that the initial point $\widehat{U}_{1:i}^0$, $\widehat{V}_{1:i}^0$ of the i-th stage, obtained using Step 4, has $c(\sigma_i^*)^2 + O\left(k(\sigma_{i+1}^*)^2\right)$ error, with $c < 1$. In the second step, we show that using the initial points $\widehat{U}_{1:i}^0$, $\widehat{V}_{1:i}^0$, the AltMin algorithm iterations in the i-th stage (Steps 6, 7) reduces the error to $\max(\epsilon, 16k\sigma_{i+1}^2)$.

Please refer [11] for a full proof of the theorem. □

7. SUMMARY AND DISCUSSION

Alternating minimization provides an empirically appealing and popular approach to solving several different low-rank matrix recovery problems. The main motivation, and result, of this paper was to provide the **first theoretical guarantees on the global optimality of alternating minimization**, for matrix completion and the related problem of matrix sensing. We would like to note the following aspects of our results and proofs:

(a): For both the problems, we show that alternating minimization recovers the true matrix under *similar problem conditions* (RIP, incoherence) to those used by existing algorithms (based on convex optimization or iterated SVDs); computationally, our results show faster convergence to the global optima, but with possibly higher sample complexity.

(b): We develop a new framework for analyzing alternating minimization for low-rank problems, where we view alternating minimization as a perturbed version of the power method. This idea is likely to have applications to other similar problems where trace-norm based convex relaxation techniques have rigorous theoretical results but alternating minimization has enjoyed more empirical success. For example, robust PCA [6, 2], spectral clustering [12] etc.

(c): Our analysis also sheds light on two key aspects of the alternating minimization approach:

Initialization: Our results show that alternating minimization succeeds if the initial iterate is not "almost orthogonal" to the target subspace. This suggests that, selecting initial iterate smartly is preferable to random initialization.

Dependence on the condition number: We show that using a stagewise adaptation of alternating minimization, we can remove the dependence on condition number for the matrix sensing problem. This suggests that stagewise versions of the basic alternating minimization algorithm may in fact perform better than the original one.

8. REFERENCES

[1] 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.

[2] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *J. ACM*, 2011.

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

[4] E. J. Candès and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203–4215, 2005.

[5] E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Trans. Inform. Theory*, 56(5):2053–2080, 2009.

[6] 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.

[7] C. Chen, B. He, and X. Yuan. Matrix completion via an alternating direction method. *IMA Journal of Numerical Analysis*, 32(1):227–245, 2012.

[8] G. H. Golub and C. F. V. Loan. *Matrix Computations*. The Johns Hopkins University Press, 1996.

[9] C.-J. Hsieh, K.-Y. Chiang, and I. S. Dhillon. Low rank modeling of signed networks. In *KDD*, 2012.

[10] P. Jain, R. Meka, and I. S. Dhillon. Guaranteed rank minimization via singular value projection. In *NIPS*, pages 937–945, 2010.

[11] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. *arXiv preprint arXiv:1212.0467*, 2012.

[12] A. Jalali, Y. Chen, S. Sanghavi, and H. Xu. Clustering partially observed graphs via convex optimization. In *ICML*, pages 1001–1008, 2011.

[13] R. H. Keshavan. Efficient algorithms for collaborative filtering. Phd Thesis, Stanford University, 2012.

[14] R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from a few entries. *IEEE Transactions on Information Theory*, 56(6):2980–2998, 2010.

[15] H. Kim and H. Park. Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method. *SIAM J. Matrix Anal. Appl.*, 30(2):713–730, July 2008.

[16] J. Kim and H. Park. Sparse nonnegative matrix factorization for clustering. Technical Report GT-CSE-08-01, Georgia Institute of Technology, 2008.

[17] Y. Koren. The BellKor solution to the Netflix grand prize, 2009.

[18] Y. Koren, R. M. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37, 2009.

[19] K. Lee and Y. Bresler. Admira: atomic decomposition for minimum rank approximation. *IEEE Transactions on Information Theory*, 56(9):4402–4416, 2010.

[20] R. Meka, P. Jain, C. Caramanis, and I. S. Dhillon. Rank minimization via online learning. In *ICML*, pages 656–663, 2008.

[21] R. Peeters. Orthogonal representations over finite fields and the chromatic number of graphs. *Combinatorica*, 16:417–431, 1996.

[22] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 2010.

[23] W. I. Zangwill. *Nonlinear Programming: A Unified Approach*. Englewood Cliffs: Prentice-Hall, 1969.

[24] Y. Zhou, D. M. Wilkinson, R. Schreiber, and R. Pan. Large-scale parallel collaborative filtering for the netflix prize. In *AAIM*, pages 337–348, 2008.

[25] H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *JCGS*, 15(2):262–286, 2006.