Fast and Sample Efficient Inductive Matrix Completion via Multi-Phase Procrustes Flow

Xiao Zhang [*1] **Simon S. Du** [*2] **Quanquan Gu** [3]

Abstract

We revisit the inductive matrix completion problem that aims to recover a rank-r matrix with ambient dimension d given n features as the side prior information. The goal is to make use of the known n features to reduce sample and computational complexities. We present and analyze a new gradient-based non-convex optimization algorithm that converges to the true underlying matrix at a linear rate with sample complexity only linearly depending on n and logarithmically depending on d. To the best of our knowledge, all previous algorithms either have a quadratic dependency on the number of features in sample complexity or a sub-linear computational convergence rate. In addition, we provide experiments on both synthetic and real world data to demonstrate the effectiveness of our proposed algorithm.

1. Introduction

Matrix completion method has been used in a wide range of applications such as collaborative filtering for recommendation (Koren et al., 2009), multi-label learning (Cabral et al., 2011) and clustering (Hsieh et al., 2012). In these applications, every entry is modeled as the inner product between factors corresponding to the row and column variables. For example, in movie recommendation, each row factor represents the latent representation of a user and each column factor represents the latent representation of a movie.

In many applications of significant interest, besides the partially observed matrix, side information, in the form of features, is also available. These might correspond to demographic information (genders, occupation) for users or product information (genre, director) in a movie recommender system for example. With such features at hand, one can model an observation as a specific linear interaction between features to reduce the model complexity. Formally, let $\mathbf{L}^* \in \mathbb{R}^{d_1 \times d_2}$ be the unknown low-rank matrix with rank r, and let $\mathbf{X}_L \in \mathbb{R}^{d_1 \times n_1}$ and $\mathbf{X}_R \in \mathbb{R}^{d_2 \times n_2}$ be the known feature matrices with $d_1 \geq n_1 \geq r$ and $d_2 \geq n_2 \geq r$. We assume the unknown rank-r matrix $\mathbf{L}^*$ can be represented by $\mathbf{X}_L \mathbf{M}^* \mathbf{X}_R^\top$ for some unknown matrix $\mathbf{M}^* \in \mathbb{R}^{n_1 \times n_2}$. Thus instead of learning a large $d_1 \times d_2$ matrix $\mathbf{L}^*$, we only need to recover a smaller low-rank matrix $\mathbf{M}^*$. This inductive approach has been applied successfully in many applications including collaborative filtering (Abernethy et al., 2009; Menon et al., 2011; Chen et al., 2012), multi-label learning (Xu et al., 2013; Si et al., 2016), semi-supervised clustering (Yi et al., 2013; Si et al., 2016), gene-disease prediction (Natarajan & Dhillon, 2014) and blog recommendation (Shin et al., 2015).

From the theoretical point of view, side information allows us to reduce the overall sample and computational complexities. Xu et al. (2013) and Jain & Dhillon (2013) pioneered the theoretical investigation in this direction. Specifically, Xu et al. (2010) adapted the convex relaxation approach (Candès & Recht, 2009; Candès & Tao, 2010) and requires only $O(rn \log n \log d)$[1] samples to recover the underlying matrix, which we believe is tight up to logarithmic factors. However, the computational cost is usually high because they need to solve a nuclear norm minimization problem, which is inherently slow due to its high per-iteration complexity and non-strongly convex objective function (c.f. Equation (2) in Xu et al. (2013)), which does not have linear convergence rate. On the other hand, Jain & Dhillon (2013) (also see Zhong et al. (2015)) proposed an algorithm which first does a spectral initialization to obtain a coarse estimate, then uses alternating minimization to refine the estimate. Their algorithm has a locally linear rate of convergence but requires $O(r^3 n^2 \log n \log(1/\epsilon))$ samples, which has an unsatisfactory quadratic dependency on n and cannot achieve exact recovery because sample complexity also depends on the target accuracy ϵ. A natural and open question is:

[*]Equal contribution [1]Department of Computer Science, University of Virginia, Charlottesville, Virginia, USA. [2]Machine Learning Department, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA. [3]Department of Computer Science, University of California, Los Angeles, CA 90095, USA. Correspondence to: Quanquan Gu <qgu@cs.ucla.edu>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

[1]For the ease of presentation, we assume $d_1 = d_2 = d$ and $n_1 = n_2 = n$ when discussing complexities.

Can we recover the ground truth matrix at a linear rate with sample complexity linear in n?

In this paper, we answer this question affirmatively. Specifically, we propose a multi-phase gradient-based algorithm that converges to the underlying true matrix at a linear rate with sample complexity linearly depending on n and logarithmically depending on d. Our algorithm is a novel and highly nontrivial extension of Procrustes Flow (Tu et al., 2015) in which we add an additional phase to reduce the variance of gradient estimate, and therefore we call it Multi-Phase Procrustes Flow. The main challenges and technical insights are summarized in the following section.

1.1. Main Challenges and Technical Insights

In recent years, a surge of non-convex optimization algorithms for estimating low-rank matrices have been established. A typical procedure is first to do a spectral initialization to obtain a coarse estimate, and then to use Burer-Monteiro factorization (Burer & Monteiro, 2003) with projected gradient descent (a.k.a., Procrustes flow) on the partially observed entries to recover the underlying matrix, where the projection is introduced to control the variance of gradient descent (Tu et al., 2015; Zheng & Lafferty, 2016; Yi et al., 2016). Our proposed algorithm also follows this framework. However, direct adaptation does not achieve the desired statistical and computational rates. Statistically, in the classical matrix completion setting, after the initialization phase, the variance of the gradient is at a smaller order than the magnitude of expected gradient for *all* iterations. However, in our setting, because of limited samples, such uniform bound does not hold. Computationally, the projection step in the inductive setting is more costly than that in the classical setting because we need to solve a convex quadratically-constrained-quadratic-programming (QCQP) problem (c.f. Section 4).

Our first key observation is that the variance of the gradient converges to 0 at a faster rate than the magnitude of expectation of the gradient. Therefore, if the iterate is close enough to the optimum, say in a ball with radius $O(1/n)$ around the optimum, the desired uniform bound still holds. Further, this observation also indicates when we are close to the optimum, projection step is *not* needed, i.e., vanilla gradient descent suffices. Nevertheless, with limited samples, the spectral initialization cannot directly achieve this goal.

Our second key observation is that after a rough spectral initialization, if we use fresh samples to calculate the gradient at each iteration, the variance is still small compared with the expectation of the gradient. In light of this, we add a new phase to the original algorithm where we use fresh samples to estimate the gradient at each iteration and use projected gradient descent to refine our estimation. Though the projection is costly, we only need $O(r \log n)$ iterations to converge to a ball with radius $O(1/n)$ around the optimum, since gradient descent in our problem enjoys a linear rate of convergence. Putting all these phases together, we propose the first gradient-based algorithm that requires only $O\left(r^2 n \log n \log d\right)$ samples and converges to the ground truth matrix at a linear rate.

Notation. Capital boldface letters such as $\mathbf{A}$ are used for matrices, and $[\ell]$ is used to denote the index set $\{1, 2, \ldots, \ell\}$. Denote the $d \times d$ identity matrix by $\mathbf{I}_d$. Let $\mathbf{A}_{i,*}$, $\mathbf{A}_{*,j}$ and A_{ij} be the i-th row, j-th column and (i,j)-th entry of matrix $\mathbf{A}$, respectively. Denote the ℓ-th largest singular value of $\mathbf{A}$ by $\sigma_\ell(\mathbf{A})$ and its projection onto the index set Ω by $\mathcal{P}_\Omega(\mathbf{A})$, i.e., the (i,j)-th entry of $\mathcal{P}_\Omega(\mathbf{A})$ is equal to A_{ij} if $(i,j) \in \Omega$ and zero otherwise. Let $\|\mathbf{x}\|_2$ be the ℓ_2 norm of a d-dimensional vector $\mathbf{x} \in \mathbb{R}^d$. Let $\|\mathbf{A}\|_F$, $\|\mathbf{A}\|_2$ be the Frobenius norm and the spectral norm of matrix $\mathbf{A}$ respectively. The largest ℓ_2 norm of its rows is defined as $\|\mathbf{A}\|_{2,\infty} = \max_i \|\mathbf{A}_{i,*}\|_2$. For any two sequences $\{a_n\}$ and $\{b_n\}$, we say $a_n = O(b_n)$ if there exists a positive constant C such that $a_n \leq C\, b_n$.

2. Related Work

2.1. Low-Rank Matrix Completion

Classical approach for matrix completion relies on convex relaxation (Candès & Recht, 2009; Candès & Tao, 2010; Recht, 2011; Chen, 2015; Allen-Zhu et al., 2017), which can be solved by nuclear norm minimization. Such methods usually have tight sample complexity (Balcan et al., 2017), but due to the use of nuclear norm and non-strongly convex objective function, they cannot achieve linear convergence rate and often scale cubically with the dimension. Some faster algorithms have been proposed (Jain & Netrapalli, 2015) but they often incur additional sample complexity.

To reduce the runtime complexity, various non-convex algorithms have been proposed. Jain et al. (2013); Hardt (2014); Hardt & Wootters (2014); Gu et al. (2016); Gamarnik et al. (2017) showed that with proper initialization, alternating minimization enjoys a linear convergence rate. Proofs of these works often build on a general analytical framework, noisy-power-method (Hardt & Price, 2014; Balcan et al., 2016). Nevertheless, the sample complexity often depends on the inverse of target accuracy. Thus these methods often cannot recover the ground truth matrix exactly.

Another line of research studies the landscape of optimization problem and showed that with proper modification of objective function, all local minima are global and all saddle points are strict (Bhojanapalli et al., 2016b; Ge et al., 2016; 2017). Therefore, perturbed gradient descent algorithms can solve this non-convex problem efficiently (Ge et al., 2015; Jin et al., 2017; Du et al., 2017a). However, to guarantee the landscape having nice properties, they all require the

sample complexity scales with the fourth power of the rank, which is suboptimal.

Lastly, Tu et al. (2015); Zhao et al. (2015); Zheng & Lafferty (2015); Sun & Luo (2015); Bhojanapalli et al. (2016a); Zheng & Lafferty (2016); Yi et al. (2016); Wang et al. (2016; 2017); Ma et al. (2017); Xu et al. (2017); Zhang et al. (2018) proposed first-order algorithms to solve low-rank matrix estimation problems. Similar to Jain et al. (2013); Hardt & Wootters (2014); Hardt (2014), these algorithms first use spectral initialization to find a good starting point, but instead of performing alternating minimization, they use (projected) gradient descent to refine the initial solution, and are guaranteed to converge to the global optimum at a linear rate. Notably, the sample complexity of these algorithms does not depend on the target accuracy and is only slightly larger than that of convex programming approaches. Our proposed algorithm also belongs to this line of research but with significant innovations in both algorithm and theory (c.f. Section 1.1).

2.2. Matrix Completion with Side Information

Matrix completion with side information has drawn much attention for improving the performance of traditional matrix completion methods in various applications. This method dates back to Jain & Dhillon (2013); Xu et al. (2013), where they proposed the so-called Inductive Matrix Completion methods independently. The method is "inductive", in that it can be generalized to previously unobserved data points, which resolves a major drawback in traditional recommender systems. Extensions to noisy features (Chiang et al., 2015) and non-linear models (Si et al., 2016) have been studied and similar formulation has also been extended to the problem of robust PCA (Chiang et al., 2016; Niranjan et al., 2017; Xue et al., 2017).

Theoretically, side information allows us to recover the target matrix with sample complexity depending on the intrinsic feature dimension rather than the ambient dimension. Information theoretically speaking, with known features, $O(rn)$ samples are sufficient for exact recovery and this is achieved up to some logarithmic factors by the convex relaxation based algorithm proposed in Xu et al. (2013). However, such formulation requires solving a nuclear norm minimization problem and in general cannot have the linear convergence. Jain & Dhillon (2013) adopted ideas from Jain et al. (2013); Hardt (2014); Hardt & Wootters (2014) to obtain a linear convergent algorithm but it requires $O\left(r^3n^2\log n\log(1/\epsilon)\right)$ samples. See Table 1 for a detailed comparison between our method and two existing inductive matrix completion algorithms: **Maxide** (Xu et al., 2013) and **AltMin**[2] (Jain & Dhillon, 2013). It is worth noting that our approach achieves both linear rate of convergence and sample complexity linear in the feature dimension n.

Table 1. Comparison results of sample complexity and convergence rate for different inductive matrix completion algorithms.

Algorithm	Sample Complexity	Linear rate?
Maxide (Xu et al., 2013)	$O\big(rn\log n\log d\big)$	No
AltMin (Jain & Dhillon, 2013)	$O\big(r^3n^2\log n\log(1/\epsilon)\big)$	Yes
Ours	$O\big(r^2n\log n\log d\big)$	Yes

3. Problem Setup and Preliminaries

Recall that our goal is to recover the unknown rank-r matrix $\mathbf{L}^*\in\mathbb{R}^{d_1\times d_2}$ by learning a lower-dimensional matrix $\mathbf{M}^*\in\mathbb{R}^{n_1\times n_2}$ given the side information in terms of $\mathbf{X}_L$ and $\mathbf{X}_R$. Denote the rank-r singular value decomposition (SVD) of $\mathbf{M}^*$ by $\mathbf{M}^*=\overline{\mathbf{U}}^*\mathbf{\Sigma}^*\overline{\mathbf{V}}^{*\top}$. Let $\sigma_1^*\geq\sigma_2^*\geq\ldots\geq\sigma_r^*>0$ be the sorted singular values of $\mathbf{M}^*$ and $\kappa=\sigma_1^*/\sigma_r^*$ be the condition number. Assume each entry of $\mathbf{L}^*$ is observed independently with probability $p\in(0,1)$. In particular, for any $(i,j)\in[d_1]\times[d_2]$, we consider the following Bernoulli observation model

$$L_{ij}=\begin{cases}L_{ij}^*, & \text{with probability } p;\\ *, & \text{otherwise.}\end{cases}\tag{3.1}$$

Let Ω be the index set of observed entries in $\mathbf{L}^*$, i.e., $\Omega=\big\{(i,j)\in[d_1]\times[d_2]\,\big|\,L_{ij}\neq *\big\}$. Note that restricting on the observed index set Ω, we have $\mathcal{P}_\Omega(\mathbf{L})=\mathcal{P}_\Omega(\mathbf{L}^*)$.

In order to fully exploit the side information, following Xu et al. (2013); Yi et al. (2013); Chiang et al. (2016), we assume the following standard feasibility condition: $\mathrm{col}(\mathbf{X}_L)\supseteq\mathrm{col}(\mathbf{L}^*)$, $\mathrm{col}(\mathbf{X}_R)\supseteq\mathrm{col}(\mathbf{L}^{*\top})$, where $\mathrm{col}(\mathbf{A})$ represents the column space of matrix $\mathbf{A}$. Intuitively, this condition suggests that the feature matrices are correlated to the underlying true low-rank space, so that we could make use of the feature information to improve our recovery. In other words, we assume $\mathbf{L}^*$ can be decomposed as $\mathbf{L}^*=\mathbf{X}_L\mathbf{M}^*\mathbf{X}_R^\top$. In addition, without loss of generality, we assume both feature matrices $\mathbf{X}_L$ and $\mathbf{X}_R$ have orthonormal columns[3], i.e., $\mathbf{X}_L^\top\mathbf{X}_L=\mathbf{I}_{n_1}$, $\mathbf{X}_R^\top\mathbf{X}_R=\mathbf{I}_{n_2}$.

It is well-known in matrix completion (Gross, 2011) that if $\mathbf{L}^*$ is equal to zero in nearly all of the rows or columns, recovering $\mathbf{L}^*$ exactly is impossible unless all of its entries

[2] Jain & Dhillon (2013) requires a weaker incoherence condition in that they only assume the features are incoherent. However, when additional incoherence condition is imposed, it is unclear whether their algorithm can reduce the sample complexity or not.

[3] In practice, one could conduct QR factorization or SVD to acquire the corresponding orthonormal feature matrices.

are sampled. Therefore, we impose the standard incoherence condition on the unknown low-rank matrix $\mathbf{L}^*$ (Candès & Recht, 2009; Recht, 2011; Yi et al., 2016). Note that given feature matrices $\mathbf{X}_L, \mathbf{X}_R$, the singular value decomposition of $\mathbf{L}^*$ can be formulated as $(\mathbf{X}_L\overline{\mathbf{U}}^*)\mathbf{\Sigma}^*(\mathbf{X}_R\overline{\mathbf{V}})^\top$.

Assumption 3.1 (Incoherence for $\mathbf{L}^*$)**.** The unknown low-rank matrix $\mathbf{L}^*$ is μ_0-incoherent, i.e., $\|\mathbf{X}_L\overline{\mathbf{U}}^*\|_{2,\infty} \leq \sqrt{\mu_0 r/d_1}$, $\|\mathbf{X}_R\overline{\mathbf{V}}^*\|_{2,\infty} \leq \sqrt{\mu_0 r/d_2}$.

Furthermore, following Jain & Dhillon (2013); Xu et al. (2013); Chiang et al. (2016), we impose the following incoherence condition on the feature matrices.

Assumption 3.2 (Incoherence for feature matrices)**.** The feature matrices $\mathbf{X}_L$ and $\mathbf{X}_R$ are both self-incoherent with parameter μ_1, i.e, $\|\mathbf{X}_L\|_{2,\infty} \leq \sqrt{\mu_1 n_1/d_1}$, $\|\mathbf{X}_R\|_{2,\infty} \leq \sqrt{\mu_1 n_2/d_2}$.

With the aid of additional feature information, inductive matrix completion can be formulated as follows

$$\begin{aligned} \min_{\mathbf{M}\in\mathbb{R}^{n_1\times n_2}} \quad & \frac{1}{2p}\big\|\mathcal{P}_\Omega(\mathbf{X}_L\mathbf{M}\mathbf{X}_R^\top - \mathbf{L})\big\|_F^2, \\ \text{subject to} \quad & \text{rank}(\mathbf{M}) \leq r, \end{aligned} \tag{3.2}$$

where Ω is the index set of observed entries and $p = |\Omega|/(d_1 d_2)$ denotes the sampling probability in the observation model. In order to estimate the low-rank matrix $\mathbf{M}^*$ more efficiently, following Tu et al. (2015), Zheng & Lafferty (2015) and Yi et al. (2016), we propose to solve the following factorized non-convex optimization problem

$$\min_{\substack{\mathbf{U}\in\mathbb{R}^{n_1\times r}\\ \mathbf{V}\in\mathbb{R}^{n_2\times r}}} \frac{1}{2p}\big\|\mathcal{P}_\Omega(\mathbf{X}_L\mathbf{U}\mathbf{V}^\top\mathbf{X}_R^\top - \mathbf{L})\big\|_F^2. \tag{3.3}$$

Due to the reparameterization $\mathbf{M} = \mathbf{U}\mathbf{V}^\top$, the rank constraint in (3.2) is automatically guaranteed in (3.3).

4. The Proposed Algorithm

Let $\mathbf{U}^* = \overline{\mathbf{U}}^*\mathbf{\Sigma}^{*1/2}$ and $\mathbf{V}^* = \overline{\mathbf{V}}^*\mathbf{\Sigma}^{*1/2}$ be the true factorized matrices. It is obvious that $(\mathbf{U}^*, \mathbf{V}^*)$ is the optimal solution to optimization problem (3.3). However, for any invertible matrix $\mathbf{P} \in \mathbb{R}^{r\times r}$, $(\mathbf{U}^*\mathbf{P}, \mathbf{V}^*(\mathbf{P}^{-1})^\top)$ is also an optimal solution. In order to deal with this identifiability issue, following Tu et al. (2015); Zheng & Lafferty (2016); Park et al. (2016), we impose an additional regularizer to the objective function in (3.3) to penalize the scale difference between $\mathbf{U}$ and $\mathbf{V}$. Specifically, we consider the following regularized optimization problem

$$\begin{aligned} \min_{\substack{\mathbf{U}\in\mathbb{R}^{n_1\times r}\\ \mathbf{V}\in\mathbb{R}^{n_2\times r}}} f_\Omega(\mathbf{U},\mathbf{V}) := & \frac{1}{2p}\big\|\mathcal{P}_\Omega(\mathbf{X}_L\mathbf{U}\mathbf{V}^\top\mathbf{X}_R^\top - \mathbf{L})\big\|_F^2 \\ & + \frac{1}{8}\big\|\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}\big\|_F^2, \end{aligned} \tag{4.1}$$

where $\mathbf{U} \in \mathbb{R}^{n_1\times r}$, $\mathbf{V} \in \mathbb{R}^{n_2\times r}$, and f_Ω denotes the regularized sample loss function. Intuitively speaking, the regularization term encourages the two factorized matrices $\mathbf{U}$ and $\mathbf{V}$ to have a similar scale.

We propose a multi-phase gradient-based algorithm to solve the proposed estimator (4.1), as shown in Algorithm 1. More specifically, we first randomly split the observed index set Ω into $S+1$ independent subsets $\{\Omega_s\}_{s=0}^S$, where Ω_0 has cardinality $|\Omega|/2$ and each of the rest has cardinality $|\Omega|/(2S)$. In Phase 1, we project the observed matrix $\mathbf{L}$ onto the first subset Ω_0 and perform rank-r SVD on $p_0^{-1}\mathcal{P}_{\Omega_0}(\mathbf{L}^*)$ to get an initial estimator $(\mathbf{U}_{\text{init}}, \mathbf{V}_{\text{init}})$, where $p_0 = |\Omega_0|/(d_1 d_2)$. We use $\text{SVD}_r(\cdot)$ to denote the rank-r SVD.

In Phase 2, we perform projected gradient descent with resampling (Jain et al., 2013; Jain & Dhillon, 2013) (a.k.a., sample splitting), where we use one fresh subset for each gradient descent update. The projection step guarantees that each intermediate iterate satisfies the similar incoherence condition as that of $(\mathbf{U}^*, \mathbf{V}^*)$, while the resampling scheme ensures the independence of the samples used in the current iteration and the previous iterates. As will be clear in the next section and in the proofs, the second phase is crucial in reducing the variance of gradient estimate and ensures the uniform convergence in the third phase. The constraint sets $\mathcal{C}_1$ and $\mathcal{C}_2$ associated with the projection are defined as

$$\begin{aligned} \mathcal{C}_1 &= \bigg\{\mathbf{U}\in\mathbb{R}^{n_1\times r} \;\Big|\; \|\mathbf{X}_L\mathbf{U}\|_{2,\infty} \leq \sqrt{\frac{\mu_0 r}{d_1}}\|\mathbf{Z}_{\text{init}}\|_2\bigg\}, \\ \mathcal{C}_2 &= \bigg\{\mathbf{V}\in\mathbb{R}^{n_2\times r} \;\Big|\; \|\mathbf{X}_R\mathbf{V}\|_{2,\infty} \leq \sqrt{\frac{\mu_0 r}{d_2}}\|\mathbf{Z}_{\text{init}}\|_2\bigg\}, \end{aligned} \tag{4.2}$$

where $\mathbf{Z}_{\text{init}}$ is specified in Phase 1. Let $\mathcal{P}_{\mathcal{C}_1}(\widehat{\mathbf{U}})$ be the projection of $\widehat{\mathbf{U}} \in \mathbb{R}^{n_1\times r}$ onto $\mathcal{C}_1$, which can be alternatively regarded as the exact solution to the following convex quadratically-constrained-quadratic-programming (QCQP)

$$\begin{aligned} &\operatorname*{argmin}_{\mathbf{U}\in\mathbb{R}^{n_1\times r}} \frac{1}{2}\|\mathbf{U} - \widehat{\mathbf{U}}\|_F^2, \\ &\text{subject to } \big\|[\mathbf{X}_L\mathbf{U}]_{i,*}\big\|_2^2 \leq \frac{\mu_0 r}{d_1}\|\mathbf{Z}_{\text{init}}\|_2^2, \forall\, i\in[d_1]. \end{aligned} \tag{4.3}$$

It is worth noting that convex QCQP problem can be solved approximately and efficiently using interior point methods (Nemirovskii, 2004). Let $\mathcal{P}_{\mathcal{C}_1}(\widehat{\mathbf{U}}, \delta)$ be the δ-approximate solution to optimization problem (4.3), i.e., $\|\mathcal{P}_{\mathcal{C}_1}(\widehat{\mathbf{U}}, \delta) - \mathcal{P}_{\mathcal{C}_1}(\widehat{\mathbf{U}})\|_F \leq \delta$. Similarly, the QCQP problem with respect to $\mathbf{V}$ is formulated in a similar way, except that $\mathbf{X}_L$ (resp. d_1) is replaced with $\mathbf{X}_R$ (resp. d_2). Accordingly, we use $\mathcal{P}_{\mathcal{C}_2}(\widehat{\mathbf{V}})$ to denote the exact projection, and $\mathcal{P}_{\mathcal{C}_2}(\widehat{\mathbf{V}}, \delta)$ to be the δ-approximate projection.

In addition, the loss function used in the s-th iteration of Phase 2 is based on the subset Ω_s, and it is identical to the loss function in (4.1), except that Ω (resp. p) is replaced with Ω_s (resp. $p_s = |\Omega_s|/(d_1 d_2)$).

Finally, in Phase 3, vanilla gradient descent is performed based on the entire observed matrix $\mathcal{P}_\Omega(\mathbf{L}^*)$. Provided these three phases, as will be seen in later analysis, Algorithm 1 is guaranteed to converge to the true factorized matrices $(\mathbf{U}^*, \mathbf{V}^*)$ with a linear rate of convergence.

Algorithm 1 GD for IMC

Input: Observed matrix $\mathcal{P}_\Omega(\mathbf{L}^*)$; feature matrices $\mathbf{X}_L$, $\mathbf{X}_R$; parameter $p_0 = |\Omega|/(2d_1d_2)$; step size τ, η; number of iterations S, T, approximation error δ.

Randomly split Ω into subsets $\Omega_0, \Omega_1, \ldots, \Omega_S$ with $|\Omega_0| = |\Omega|/2$ and $|\Omega_s| = |\Omega|/(2S)$, for any $s \in [S]$

// Phase 1: Initialization

$[\widetilde{\mathbf{U}}_0, \mathbf{\Sigma}_0, \widetilde{\mathbf{V}}_0] = \text{SVD}_r\big(p_0^{-1}\mathcal{P}_{\Omega_0}(\mathbf{L}^*)\big)$

$\mathbf{U}_{\text{init}} = \mathbf{X}_L^\top \widetilde{\mathbf{U}}_0 \mathbf{\Sigma}_0^{1/2}$; $\mathbf{V}_{\text{init}} = \mathbf{X}_R^\top \widetilde{\mathbf{V}}_0 \mathbf{\Sigma}_0^{1/2}$

$\mathbf{Z}_{\text{init}} = [\mathbf{U}_{\text{init}}; \mathbf{V}_{\text{init}}]$

// Phase 2: PGD with subsamples

$\mathbf{U}_0 = \mathcal{P}_{\mathcal{C}_1}(\mathbf{U}_{\text{init}}, \delta), \mathbf{V}_0 = \mathcal{P}_{\mathcal{C}_2}(\mathbf{V}_{\text{init}}, \delta)$

for: $s = 1, 2, \ldots, S$ **do**

$\mathbf{U}_s = \mathcal{P}_{\mathcal{C}_1}\big(\mathbf{U}_{s-1} - \eta \nabla_{\mathbf{U}} f_{\Omega_s}(\mathbf{U}_{s-1}, \mathbf{V}_{s-1}), \delta\big)$

$\mathbf{V}_s = \mathcal{P}_{\mathcal{C}_2}\big(\mathbf{V}_{s-1} - \eta \nabla_{\mathbf{V}} f_{\Omega_s}(\mathbf{U}_{s-1}, \mathbf{V}_{s-1}), \delta\big)$

end for

// Phase 3: Vanilla GD

$\mathbf{U}^0 = \mathbf{U}_S, \mathbf{V}^0 = \mathbf{V}_S$

for: $t = 0, 1, \ldots, T-1$ **do**

$\mathbf{U}^{t+1} = \mathbf{U}^t - \tau \nabla_{\mathbf{U}} f_\Omega(\mathbf{U}^t, \mathbf{V}^t)$

$\mathbf{V}^{t+1} = \mathbf{V}^t - \tau \nabla_{\mathbf{V}} f_\Omega(\mathbf{U}^t, \mathbf{V}^t)$

end for

Output: $(\mathbf{U}^T, \mathbf{V}^T)$

5. Main Theory

Before presenting the main theoretical results, we note that the optimal solution to optimization problem (4.1) is not unique. Therefore, following Tu et al. (2015), we introduce the so-called Procrustes distance. For simplicity, we let $\mathbf{Z}^* = [\mathbf{U}^*; \mathbf{V}^*]$ be the stacked true parameter matrix.

Definition 5.1. For any $\mathbf{Z} \in \mathbb{R}^{(n_1+n_2)\times r}$, let $D(\mathbf{Z}, \mathbf{Z}^*)$ be the minimal distance between $\mathbf{Z}$ and $\mathbf{Z}^*$ in terms of the optimal rotation, or more precisely, $D(\mathbf{Z}, \mathbf{Z}^*) = \min_{\mathbf{R}\in\mathbb{Q}_r} \|\mathbf{Z} - \mathbf{Z}^*\mathbf{R}\|_F$, where $\mathbb{Q}_r$ denotes the set of r-by-r othorgonal matrices.

In the following discussions, we use d and n to denote $\max\{d_1, d_2\}$ and $\max\{n_1, n_2\}$, respectively. Our main theoretical result on Algorithm 1 is presented as follows.

Theorem 5.2. Assume the observed index set Ω follows Bernoulli model (3.1) and incoherence Assumptions 3.1, 3.2 hold. There exist constants c_1, c_2, c_3, c_4, c_5 such that under condition $|\Omega| \geq c_1 \max\{\mu_1 n, \mu_0 r\kappa\}\mu_0 r^2\kappa^2 \log n \log d$, if step size $\eta = c_2/(r\sigma_1^*)$, $\tau = c_3/\sigma_1^*$ and approximation error $\delta = O\big(1/(r\kappa n^2)\big)$, after $S = O(r\kappa \log n)$ iterations in Phase 2 and $T = O\left(\kappa \log(1/\epsilon)\right)$ iterations in Phase 3, with probability at least $1 - c_4 r\kappa \log n/d$, the output of Algorithm 1 satisfies

$$\|\mathbf{M}^T - \mathbf{M}^*\|_F \leq c_5\sqrt{\sigma_1^*}\epsilon,$$

where $\mathbf{M}^T = \mathbf{U}^T\mathbf{V}^{T\top}$ and $\mathbf{M}^* = \mathbf{U}^*\mathbf{V}^{*\top}$.

Theorem 5.2 shows that the overall sample complexity of Algorithm 1 is $O\big(r^2\kappa^2 n \log n \log d\big)$. Here, we explicitly write down the dependency on condition number κ in the $O(\cdot)$ notation for completeness. It is worth noting that our gradient-based Algorithm 1 achieves both linear rate of convergence and sample complexity linearly depending on n, compared with convex relaxation based approach (Xu et al., 2013) whose convergence rate is sublinear (i.e., $O(1/\sqrt{\epsilon})$) and alternation minimization (Jain & Dhillon, 2013), which requires at least $O\big(r^3n^2 \log n \log(1/\epsilon)\big)$ samples.

Theorem 5.2 can be achieved by analyzing the three phases of Algorithm 1. In the sequel, we are going to provide the theoretical guarantees of each phase.

Theorem 5.3 (Initialization)**.** Assume the observed index set Ω follows Bernoulli model (3.1). Suppose Assumptions 3.1, 3.2 hold for the unknown low-rank matrix $\mathbf{L}^*$ and the feature matrices $\mathbf{X}_L, \mathbf{X}_R$, respectively. For any $\gamma \in (0, 1)$, there exist constants c_1, c_2 such that under the condition $|\Omega_0| \geq c_1\mu_0\mu_1 r^2\kappa^2 n \log d/\gamma^2$, with probability at least $1 - c_2/d$, the output of Phase 1 in Algorithm 1 satisfies

$$D(\mathbf{Z}_{\text{init}}, \mathbf{Z}^*) \leq 4\gamma\sqrt{\sigma_r^*}.$$

Theorem 5.3 suggests that the output of Initialization Phase 1 is already in a small neighbourhood of the optimum with radius $O(\sqrt{\sigma_r^*})$. Notably, the sample complexity is linear in n, in sharp contrast to that of the classical matrix completion setting which is at least linear in d.

Theorem 5.4 (PGD with subsamples)**.** Under the same conditions as in Theorem 5.3, suppose the output of Phase 1, $\mathbf{Z}_{\text{init}}$, satisfies $D(\mathbf{Z}_{\text{init}}, \mathbf{Z}^*) \leq \alpha\sqrt{\sigma_r^*}/2$ with constant $\alpha \leq 1/40$. There exist constants c_1, c_2, c_3, c_4, c_5 such that, if the total sample size $|\Omega| \geq c_1 S \cdot \max\{\mu_0\mu_1 r\kappa n, \mu_0^2 r^2\kappa^2\} \log d$, with step size $\eta = c_2/(r\sigma_1^*)$ and approximation error $\delta \leq c_3\sqrt{\sigma_r^*}/(r\kappa)$, the final iterate $(\mathbf{U}_S, \mathbf{V}_S)$ in Phase 2 of Algorithm 1 satisfies

$$D^2(\mathbf{Z}_S, \mathbf{Z}^*) \leq \left(1 - \frac{c_2}{16r\kappa}\right)^S \alpha^2\sigma_r^* + c_4\delta r\kappa\sqrt{\sigma_r^*} \quad (5.1)$$

with probability at least $1 - c_5 S/d$, where $\mathbf{Z}_S = [\mathbf{U}_S; \mathbf{V}_S]$.

The last term on the right hand side of (5.1) originates from the approximation error δ when solving the convex QCQP (4.3) with respect to $\mathbf{U}$ (or $\mathbf{V}$). Theorem 5.4 suggests that under proper initialization, the gradient iteration in Phase 2 converges at a linear rate with contraction parameter $1 -$

$O(1/(r\kappa))$. Note that the step size is chosen as $O(1/(r\sigma_1^*))$. In practice, since σ_1^* is unknown, we can approximate σ_1^* by $C \cdot \|\mathbf{U}_{\text{init}}\mathbf{V}_{\text{init}}^\top\|_2$ and tune the coefficient C.

Theorem 5.5 (Vanilla GD)**.** Under the same conditions as in Theorem 5.3, suppose the final iterate $(\mathbf{U}_S, \mathbf{V}_S)$ of Phase 2 in Algorithm 1 satisfies $D(\mathbf{Z}_S, \mathbf{Z}^*) \leq c_0\sqrt{\sigma_r^*}/(\mu_1 n)$ with $\mathbf{Z}_S = [\mathbf{U}_S; \mathbf{V}_S]$ and constant c_0 small enough. Then there exist constants c_1, c_2, c_3 such that if $|\Omega| \geq c_1\mu_0\mu_1 rn\log d$, with step size $\tau = c_2/\sigma_1^*$, the output of Phase 3 satisfies

$$D^2(\mathbf{Z}^T, \mathbf{Z}^*) \leq \left(1 - \frac{\tau\sigma_r^*}{16}\right)^T D^2(\mathbf{Z}_S, \mathbf{Z}^*)$$

with probability at least $1 - c_3/d$, where $\mathbf{Z}^T = [\mathbf{U}^T; \mathbf{V}^T]$.

Theorem 5.5 implies that if the final iterate of Phase 2 falls into a even smaller neighbourhood around the optimum with radius $O(1/n)$, vanilla gradient descent suffices to guarantee the linear rate of convergence.

Remark 5.6 (Computational Complexity)**.** The rank-r SVD in Phase 1 requires $O(r|\Omega_0|)$ computation. The runtime of the gradient computation for the s-th iteration in the second phase is $O(rn|\Omega_s| + r^2 n)$, while solving the convex QCQP subproblem requires $O(r^2n^2d^{3/2}\log d)$ computation if using the path-following interior point method (Nemirovskii, 2004). Thus to perform $S = O(r\kappa\log n)$ iterations, the overall computational complexity of Phase 2 is $O(r^3n^2d^{3/2}\log n\log d)$. The runtime of gradient computation in each iteration of Phase 3 is $O(rn|\Omega|)$ and the total number of iterations required in Phase 3 is $T = O(\kappa\log(1/\epsilon))$, which implies the overall computational cost in Phase 3 is $O(r^3n^2\log n\log d\log(1/\epsilon))$. Putting all these pieces together, we conclude the total computational complexity of Algorithm 1 is $O(r^3n^2\log n\log d\log(1/\epsilon) + r^3n^2d^{3/2}\log n\log d)$.

6. Experiments

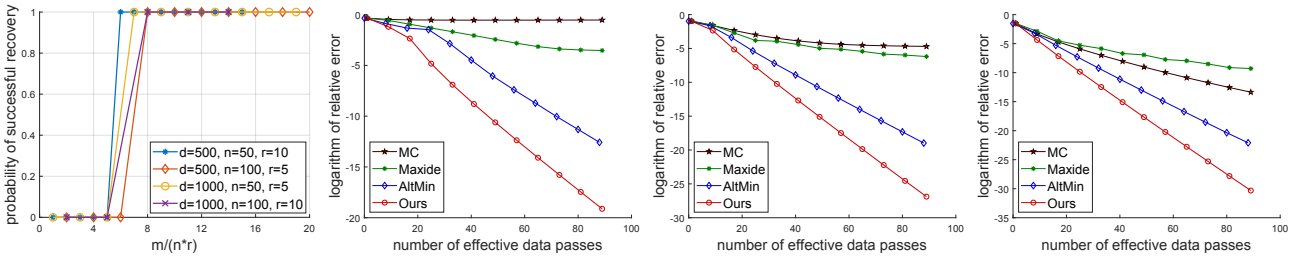
In this section, we compare the proposed gradient-based algorithm with existing inductive matrix completion methods, including the convex relaxation based approach, **Maxide** (Xu et al., 2013) and alternating minimization based algorithm, **AltMin** (Jain & Dhillon, 2013) on both synthetic and real datasets. In addition, the standard matrix completion approach based on non-convex projected gradient descent (Zheng & Lafferty, 2016) (**MC**) is compared as a baseline for simulations and the second real data experiment on gene-disease prediction, while the Binary Relevance approach (Boutell et al., 2004) using linear kernel SVM (Chang & Lin, 2011) (**BR-linear**) is included as a baseline for the first real data experiment on multi-label learning. All algorithms are implemented in Matlab on a machine with Intel 8-core Core i7 3.40 GHz with 8GB RAM.

6.1. Simulations

For simplicity, we choose $d_1 = d_2 = d$ and $n_1 = n_2 = n$. Additional experiments regarding the rectangular setting are postponed to the supplemental materials. The unknown low-rank matrix $\mathbf{M}^* \in \mathbb{R}^{n\times n}$ is generated such that $\mathbf{M}^* = \mathbf{U}^*\mathbf{V}^{*\top}$, and the entries of $\mathbf{U}^*, \mathbf{V}^* \in \mathbb{R}^{n\times r}$ are drawn independently from centered Gaussian distribution with variance $1/n$. Let the singular value decomposition of a random matrix $\mathbf{F} \in \mathbb{R}^{d\times d}$ be $\mathbf{F} = \mathbf{Y}_L\mathbf{\Sigma}\mathbf{Y}_R^\top$, where each entry of $\mathbf{F}$ is drawn independently from standard normal distribution. The feature matrices $\mathbf{X}_L, \mathbf{X}_R \in \mathbb{R}^{d\times n}$ are then generated as the first n columns of the singular matrices $\mathbf{Y}_L$ and $\mathbf{Y}_R$ respectively. The observed data matrix $\mathbf{L}$ follows from the Bernoulli model (3.1) with the full data matrix defined by $\mathbf{L}^* = \mathbf{X}_L\mathbf{M}^*\mathbf{X}_R^\top$.

To begin with, we investigate the sample complexity of the proposed gradient-based method. In particular, we consider the following settings: (i) $d = 500, n = 50, r = 10$; (ii) $d = 500, n = 100, r = 5$; (iii) $d = 1000, n = 50, r = 5$; (iv) $d = 1000, n = 100, r = 10$. We compute the empirical probability of successful recovery after 50 repeated trials, where we regard the trial as successful if the relative error $\|\mathbf{X}_L\mathbf{M}^T\mathbf{X}_R^\top - \mathbf{L}^*\|_F/\|\mathbf{L}^*\|_F$ is less than 10^{-6}. The experimental results are shown in Figure 1(a). Here, m represents the total number of observed entries. Under all of the aforementioned settings, the phase transition happens to be around $m/(nr) = 6$, which implies that the optimal sample complexity for gradient-based inductive matrix completion approach may be linear in both n and r.

Moreover, we compare our algorithm with the aforementioned algorithms, including **MC**, **Maxide** and **AltMin**. All the parameters, such as step size and regularization parameters, are tuned by 5-fold cross validation. We measure the performance by the relative reconstruction error $\|\widehat{\mathbf{L}} - \mathbf{L}^*\|_F/\|\mathbf{L}^*\|_F$ under the setting that $d = 1000$, $n = 100$, $r = 10$ with sampling rate p varied in the range $\{2\%, 5\%, 10\%\}$. For the sake of fairness, we use the same initialization procedure as in Algorithm 1 for all the compared algorithms. The results are demonstrated in Figures 1(b), 1(c) and 1(d). Here, each effective data pass evaluates $|\Omega|$ observed entries. It can be seen that inductive methods can recover the unknown low-rank matrix $\mathbf{L}^*$ successfully using less observed entries compared with the standard matrix completion approach, which proves the effectiveness of feature information. In addition, our approach achieves the lowest recovery error with respect to the same number of effective data passes, and outperforms existing inductive matrix completion algorithms by a large margin. In addition, we also plot the relative error with respect to CPU time, and similar trend in results can be observed. Due to space limit, these plots are deferred to the supplementary materials. All these comparison results clearly demonstrate the superiority



(a) sample complexity (b) $p = 2\%$ (c) $p = 5\%$ (d) $p = 10\%$

Figure 1. Experimental results on synthetic datasets: (a) Plot of empirical probability of successful recovery versus $m/(nr)$ based on our proposed algorithm. (b),(c) and (d) Plots of logarithm relative error versus effective data passes for different (inductive) matrix completion algorithms under the setting $d = 1000$, $n = 100$ and $r = 10$ with different sampling rate p.

Table 2. Experimental results in terms of AP and total running time on NUS-WIDE-OBJECT dataset for multi-label learning via different methods. p represents the percentage of observed instances. The best averaged AP (the higher the better) is bolded for each setting.

Dataest	Method	$p = 10\%$		$p = 25\%$		$p = 50\%$	
		averaged AP (std)	time (s)	averaged AP (std)	time (s)	averaged AP (std)	time (s)
NUS-WIDE OBJECT	BR-linear	0.3280 (0.0037)	9.72×10^2	0.3357 (0.0046)	2.77×10^3	0.3428 (0.0031)	7.15×10^3
	Maxide	0.5349 (0.0034)	3.21×10^1	0.5562 (0.0021)	3.42×10^1	0.5629 (0.0023)	3.27×10^1
	AltMin	0.5265 (0.0031)	1.92×10^1	0.5536 (0.0028)	2.11×10^1	0.5591 (0.0027)	2.03×10^1
	Ours	**0.5434** (0.0040)	7.53×10^0	**0.5677** (0.0027)	7.19×10^0	**0.5718** (0.0023)	9.57×10^0

of our proposed algorithm in terms of computation and is well aligned with our theory.

6.2. Multi-Label Learning

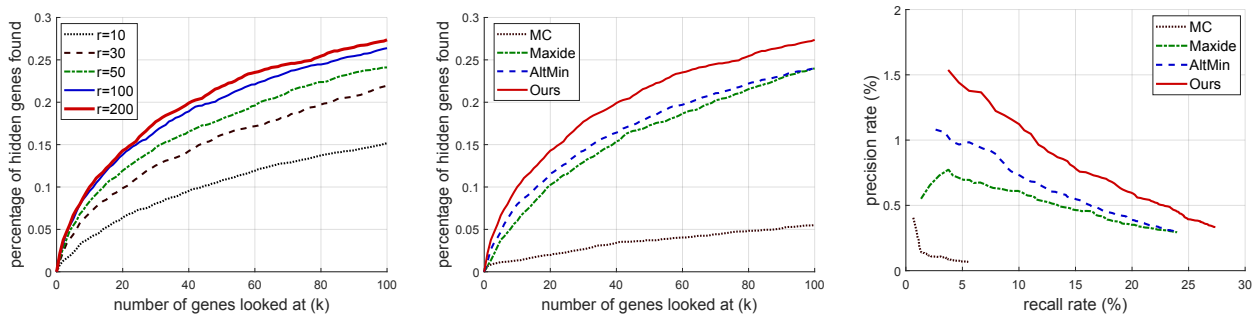
We also apply our proposed algorithm to multi-label learning on the image classification dataset **NUS-WIDE-OBJECT** obtained from Chua et al. (2009), which is one of the prominent applications of inductive matrix completion. Additional experiments on Yahoo datasets (Ueda & Saito, 2003) are deferred to the supplementary materials. The NUS-WIDE-OBJECT dataset consists of $d_1 = 30000$ images classified by $d_2 = 31$ object categories, along with 5 types of low-level features extracted from these images. We construct the feature matrix by further extracting the top-50 principle components from each type of side information, which leads to $n_1 = 250$ features in total. Detailed information regarding the dataset can be found in Chua et al. (2009). Our goal is to predict the labels associated with the unseen instances, based on both the side information as well as the label assignments of the observed instances. By leveraging the low-rankness property of the unknown instance-label matrix (Ji et al., 2008; Goldberg et al., 2010), multi-label learning can be reformulated as an inductive matrix completion problem (3.2), where $\mathbf{L}^*$ is the instance-label matrix, $\mathbf{X}_L$ represents the feature matrix and $\mathbf{X}_R$ is set as an identity matrix in this context.

We randomly sample $p \times 100\%$ instances as the observed (training) data for each dataset, and treat the remaining $(1-p) \times 100\%$ instances as the unobserved (testing) data, with p chosen from $\{10\%, 25\%, 50\%\}$. We estimate the unknown matrix of parameters based on the training data, and report the average precision (AP) (Zhang & Zhou, 2014) computed from the testing data. Specifically, the average precision measures the averaged fraction of relevant labels ranked higher than a specific label. We compare our algorithm with the baseline approach, **BR-linear**, and existing inductive matrix completion algorithms, **Maxide** and **AltMin**. All the parameters, including the rank r (we tune it over the grid $\{5, 10, \ldots, 30\}$), are tuned via 5-fold cross validation based on the training data. Table 2 depicts the detailed experimental results. In detail, for each setting of observed training data, we report the averaged AP over 10 trials and the corresponding standard deviation as well as the total run time. We can observe from Table 2 that the proposed gradient-based algorithm outperforms the BR-linear by a large margin. Compared with existing inductive matrix completion algorithms, our algorithm also achieves significantly better results under all of the experimental settings in terms of both prediction accuracy and running time. This again illustrates the advantage of our algorithm.

6.3. Gene-Disease Prediction

We further apply our proposed method for predicting gene-disease associations on the OMIM[4] data used in Singh-Blom et al. (2013), which is another successful application of inductive matrix completion. In the context of gene-disease

[4]OMIM is short for Online Mendelian Inheritance in Man, which is a public database for human gene-disease studies.



(a) performance for different r (b) CDF of the rankings (c) precision-recall curves

Figure 2. Experimental results for predicting gene-disease associations: (a) Plot of the probability that a true gene-disease association is recovered in the top-k predictions based our proposed method for different rank r; (b) Comparisons of different (inductive) matrix completion methods based on the empirical cumulative distribution of the rankings with rank $r = 200$; (c) Comparisons of different (inductive) matrix completion methods with respect to the standard precision recall measures with rank $r = 200$.

association prediction, we let $\mathbf{L}^* \in \mathbb{R}^{d_1 \times d_2}$ be the gene-disease association matrix, such that $L^*_{ij} = 1$ if gene i is associated with disease j; $L^*_{ij} = 0$ if the association is unobserved. On this dataset, the association matrix $\mathbf{L}^*$ is highly sparse, consisting of $d_1 = 12331$ different genes and $d_2 = 3209$ different diseases with only 3954 discovered gene-disease associations. In addition, we obtain the gene feature matrix $\mathbf{X}_L \in \mathbb{R}^{d_1 \times n_1}$ and disease feature matrix $\mathbf{X}_R \in \mathbb{R}^{d_2 \times n_2}$ from Natarajan & Dhillon (2014), where $n_1 = 300$ gene features and $n_2 = 200$ disease features are extracted respectively. Our objective is to predict potential genes for certain diseases of interest based on both the observed associations and feature information, which can thus be formulated as an inductive matrix completion problem. Following Natarajan & Dhillon (2014), we include an additional regularization term in (4.1) to take into account the sparsity of the underlying association matrix

$$\min_{\substack{\mathbf{U}\in\mathbb{R}^{n_1\times r}\\ \mathbf{V}\in\mathbb{R}^{n_2\times r}}} f_\Omega(\mathbf{U},\mathbf{V}) + \lambda \left\|\mathcal{P}_{\Omega^c}(\mathbf{X}_L\mathbf{U}\mathbf{V}^\top\mathbf{X}_R^\top)\right\|_F^2, \quad (6.1)$$

where r is the supposed rank of $\mathbf{L}^*$, Ω stands for the (training) index set of gene-disease associations, and Ω^c represents its complement. Note that all the algorithms we studied here including ours can be directly applied to solve (6.1) with slight modification. In our experiment, we tune the regularization parameter λ via cross validation and choose the best value $\lambda = 0.5$.

To evaluate the performance of our method, we equally split the known gene-disease associations into three groups and perform 3-fold cross validation. Specifically, we treat each group as testing data once and apply our gradient-based method on the remaining two groups to obtain the estimation matrix of $\mathbf{L}^*$. For every gene-disease pair (g, d) in the testing group, we order all the genes by the corresponding estimated values associated with disease d, and then record the ranking of gene g in the list. We use the cumulative distribution of the rankings (Singh-Blom et al., 2013; Natarajan & Dhillon, 2014) as the performance measure for evaluation, i.e., the probability that the ranking is less than a specific threshold $k \in \{1, 2, \ldots, 100\}$. The experimental results with rank r varied in the range $\{10, 30, 50, 100, 200\}$ based on our method are displayed in Figure 2(a), which indicates that the rank plays an important role in gene-disease prediction: higher rank leads to better performance. In later experiments, we choose $r = 200$ because the performance of inductive matrix completion on this dataset tends to be saturated when $r = 200$.

Moreover, we compare our algorithm with the following algorithms: **MC**, **Maxide** and **AltMin**, which are discussed at the beginning of Section 6. The comparison results in terms of the cumulative distribution of the rankings are illustrated in Figure 2(b). It can be seen that our proposed algorithm uniformly outperforms other methods over all threshold values k. In addition, we present the precision-recall curves for all the methods we compared in Figure 2(c). Here the precision is defined as the ratio of true recovered gene-disease associations to the total number of associations we assessed; and the recall is the fraction of the true gene-disease associations that are recovered. Again, the proposed method dominates other relevant approaches, which suggests that our method can better serve for biologists to discover new gene-disease associations.

7. Conclusions and Future Work

In this paper we proposed the first gradient-based non-convex optimization algorithm for inductive matrix completion with sample complexity linear in the number of features and converges to the unknown low-rank matrix at a linear rate. One possible future direction is to extend our algorithm to the case with noisy side information (Chiang et al., 2015) or the agnostic setting, i.e., the underlying matrix has high rank (Du et al., 2017b). Another direction is to generalize our approach to non-linear models (Si et al., 2016).

Acknowledgments

We would like to thank the anonymous reviewers and Yining Wang for their helpful comments. XZ and QG are partially supported by the National Science Foundation IIS-1618948, IIS-1652539, IIS-1717206 and BIGDATA IIS-1741342, SD is partially supported by NSF grant IIS-1563887, AFRL grant FA8750-17-2-0212 and DARPA D17AP00001.

References

Abernethy, J., Bach, F., Evgeniou, T., and Vert, J.-P. A new approach to collaborative filtering: Operator estimation with spectral regularization. *Journal of Machine Learning Research*, 10(Mar):803–826, 2009.

Allen-Zhu, Z., Hazan, E., Hu, W., and Li, Y. Linear convergence of a frank-wolfe type algorithm over trace-norm balls. *arXiv preprint arXiv:1708.02105*, 2017.

Balcan, M.-F., Du, S. S., Wang, Y., and Yu, A. W. An improved gap-dependency analysis of the noisy power method. In *Conference on Learning Theory*, pp. 284–309, 2016.

Balcan, M.-F., Liang, Y., Woodruff, D. P., and Zhang, H. Optimal sample complexity for matrix completion and related problems via l_2-regularization. *arXiv preprint arXiv:1704.08683*, 2017.

Bhojanapalli, S., Kyrillidis, A., and Sanghavi, S. Dropping convexity for faster semi-definite optimization. In *Conference on Learning Theory*, pp. 530–582, 2016a.

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

Boutell, M. R., Luo, J., Shen, X., and Brown, C. M. Learning multi-label scene classification. *Pattern recognition*, 37(9): 1757–1771, 2004.

Burer, S. and Monteiro, R. D. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming*, 95(2):329–357, 2003.

Cabral, R. S., Torre, F., Costeira, J. P., and Bernardino, A. Matrix completion for multi-label image classification. In *Advances in Neural Information Processing Systems*, pp. 190–198, 2011.

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

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

Chang, C.-C. and Lin, C.-J. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):27, 2011.

Chen, T., Zhang, W., Lu, Q., Chen, K., Zheng, Z., and Yu, Y. Svdfeature: a toolkit for feature-based collaborative filtering. *Journal of Machine Learning Research*, 13(Dec):3619–3622, 2012.

Chen, Y. Incoherence-optimal matrix completion. *IEEE Transactions on Information Theory*, 61(5):2909–2923, 2015.

Chiang, K.-Y., Hsieh, C.-J., and Dhillon, I. S. Matrix completion with noisy side information. In *Advances in Neural Information Processing Systems*, pp. 3447–3455, 2015.

Chiang, K.-Y., EDU, U., Hsieh, C.-J., and Dhillon, E. I. S. Robust principal component analysis with side information. In *Proceedings of The 33rd International Conference on Machine Learning*, pp. 2291–2299, 2016.

Chua, T.-S., Tang, J., Hong, R., Li, H., Luo, Z., and Zheng, Y. Nus-wide: a real-world web image database from national university of singapore. In *Proceedings of the ACM international conference on image and video retrieval*, pp. 48. ACM, 2009.

Du, S. S., Jin, C., Lee, J. D., Jordan, M. I., Poczos, B., and Singh, A. Gradient descent can take exponential time to escape saddle points. *arXiv preprint arXiv:1705.10412*, 2017a.

Du, S. S., Wang, Y., and Singh, A. On the power of truncated svd for general high-rank matrix estimation problems. *arXiv preprint arXiv:1702.06861*, 2017b.

Gamarnik, D., Li, Q., and Zhang, H. Matrix completion from $o(n)$ samples in linear time. *arXiv preprint arXiv:1702.02267*, 2017.

Ge, R., Huang, F., Jin, C., and Yuan, Y. Escaping from saddle pointsonline stochastic gradient for tensor decomposition. In *Conference on Learning Theory*, pp. 797–842, 2015.

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

Ge, R., Jin, C., and Zheng, Y. No spurious local minima in nonconvex low rank problems: A unified geometric analysis. *arXiv preprint arXiv:1704.00708*, 2017.

Goldberg, A., Recht, B., Xu, J., Nowak, R., and Zhu, X. Transduction with matrix completion: Three birds with one stone. In *Advances in neural information processing systems*, pp. 757–765, 2010.

Gross, D. Recovering low-rank matrices from few coefficients in any basis. *IEEE Transactions on Information Theory*, 57(3): 1548–1566, 2011.

Gu, Q., Wang, Z. W., and Liu, H. Low-rank and sparse structure pursuit via alternating minimization. In *Artificial Intelligence and Statistics*, pp. 600–609, 2016.

Hardt, M. Understanding alternating minimization for matrix completion. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pp. 651–660. IEEE, 2014.

Hardt, M. and Price, E. The noisy power method: A meta algorithm with applications. In *Advances in Neural Information Processing Systems*, pp. 2861–2869, 2014.

Hardt, M. and Wootters, M. Fast matrix completion without the condition number. In *Conference on Learning Theory*, pp. 638–678, 2014.

Hsieh, C.-J., Chiang, K.-Y., and Dhillon, I. S. Low rank modeling of signed networks. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 507–515. ACM, 2012.

Jain, P. and Dhillon, I. S. Provable inductive matrix completion. *arXiv preprint arXiv:1306.0626*, 2013.

Jain, P. and Netrapalli, P. Fast exact matrix completion with finite samples. In *COLT*, pp. 1007–1034, 2015.

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

Ji, S., Tang, L., Yu, S., and Ye, J. Extracting shared subspace for multi-label classification. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 381–389. ACM, 2008.

Jin, C., Ge, R., Netrapalli, P., Kakade, S. M., and Jordan, M. I. How to escape saddle points efficiently. *arXiv preprint arXiv:1703.00887*, 2017.

Koren, Y., Bell, R., and Volinsky, C. Matrix factorization techniques for recommender systems. *Computer*, 42(8), 2009.

Ma, C., Wang, K., Chi, Y., and Chen, Y. Implicit regularization in nonconvex statistical estimation: Gradient descent converges linearly for phase retrieval, matrix completion and blind deconvolution. *arXiv preprint arXiv:1711.10467*, 2017.

Menon, A. K., Chitrapura, K.-P., Garg, S., Agarwal, D., and Kota, N. Response prediction using collaborative filtering with hierarchies and side-information. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 141–149. ACM, 2011.

Natarajan, N. and Dhillon, I. S. Inductive matrix completion for predicting gene–disease associations. *Bioinformatics*, 30(12): i60–i68, 2014.

Nemirovskii, A. Interior point polynomial time methods in convex programming, 2004. *Lecture Notes*, 2004.

Niranjan, U., Rajkumar, A., and Tulabandhula, T. Provable inductive robust pca via iterative hard thresholding. *arXiv preprint arXiv:1704.00367*, 2017.

Park, D., Kyrillidis, A., Caramanis, C., and Sanghavi, S. Finding low-rank solutions to matrix problems, efficiently and provably. *arXiv preprint arXiv:1606.03168*, 2016.

Recht, B. A simpler approach to matrix completion. *Journal of Machine Learning Research*, 12(Dec):3413–3430, 2011.

Shin, D., Cetintas, S., Lee, K.-C., and Dhillon, I. S. Tumblr blog recommendation with boosted inductive matrix completion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pp. 203–212. ACM, 2015.

Si, S., Chiang, K.-Y., Hsieh, C.-J., Rao, N., and Dhillon, I. S. Goal-directed inductive matrix completion. In *KDD*, pp. 1165–1174, 2016.

Singh-Blom, U. M., Natarajan, N., Tewari, A., Woods, J. O., Dhillon, I. S., and Marcotte, E. M. Prediction and validation of gene-disease associations using methods inspired by social network analyses. *PloS one*, 8(5):e58977, 2013.

Sun, R. and Luo, Z.-Q. Guaranteed matrix completion via nonconvex factorization. In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pp. 270–289. IEEE, 2015.

Tropp, J. A. User-friendly tail bounds for sums of random matrices. *Foundations of computational mathematics*, 12(4):389–434, 2012.

Tu, S., Boczar, R., Simchowitz, M., Soltanolkotabi, M., and Recht, B. Low-rank solutions of linear matrix equations via procrustes flow. *arXiv preprint arXiv:1507.03566*, 2015.

Ueda, N. and Saito, K. Parametric mixture models for multi-labeled text. In *Advances in neural information processing systems*, pp. 737–744, 2003.

Wang, L., Zhang, X., and Gu, Q. A unified computational and statistical framework for nonconvex low-rank matrix estimation. *arXiv preprint arXiv:1610.05275*, 2016.

Wang, L., Zhang, X., and Gu, Q. A universal variance reduction-based catalyst for nonconvex low-rank matrix recovery. *arXiv preprint arXiv:1701.02301*, 2017.

Xu, H., Caramanis, C., and Sanghavi, S. Robust pca via outlier pursuit. In *Advances in Neural Information Processing Systems*, pp. 2496–2504, 2010.

Xu, M., Jin, R., and Zhou, Z.-H. Speedup matrix completion with side information: Application to multi-label learning. In *Advances in NIPS*, pp. 2301–2309, 2013.

Xu, P., Ma, J., and Gu, Q. Speeding up latent variable gaussian graphical model estimation via nonconvex optimization. In *Advances in Neural Information Processing Systems*, pp. 1930–1941, 2017.

Xue, N., Deng, J., Panagakis, Y., and Zafeiriou, S. Informed non-convex robust principal component analysis with features. *arXiv preprint arXiv:1709.04836*, 2017.

Yi, J., Zhang, L., Jin, R., Qian, Q., and Jain, A. Semi-supervised clustering by input pattern assisted pairwise similarity matrix completion. In *International Conference on Machine Learning*, pp. 1400–1408, 2013.

Yi, X., Park, D., Chen, Y., and Caramanis, C. Fast algorithms for robust pca via gradient descent. In *Advances in neural information processing systems*, pp. 4152–4160, 2016.

Zhang, M.-L. and Zhou, Z.-H. A review on multi-label learning algorithms. *IEEE transactions on knowledge and data engineering*, 26(8):1819–1837, 2014.

Zhang, X., Wang, L., and Gu, Q. A unified framework for nonconvex low-rank plus sparse matrix recovery. In *International Conference on Artificial Intelligence and Statistics*, pp. 1097–1107, 2018.

Zhao, T., Wang, Z., and Liu, H. A nonconvex optimization framework for low rank matrix estimation. In *Advances in Neural Information Processing Systems*, pp. 559–567, 2015.

Zheng, Q. and Lafferty, J. A convergent gradient descent algorithm for rank minimization and semidefinite programming from random linear measurements. In *Advances in Neural Information Processing Systems*, pp. 109–117, 2015.

Zheng, Q. and Lafferty, J. Convergence analysis for rectangular matrix completion using burer-monteiro factorization and gradient descent. *arXiv preprint arXiv:1605.07051*, 2016.

Zhong, K., Jain, P., and Dhillon, I. S. Efficient matrix sensing using rank-1 gaussian measurements. In *International Conference on Algorithmic Learning Theory*, pp. 3–18. Springer, 2015.