Unique sparse decomposition of low rank matrices

Anonymous Author(s)
Affiliation
Address
email

Abstract

The problem of finding the unique low dimensional decomposition of a given matrix has been a fundamental and recurrent problem in many areas. In this paper, we study the problem of seeking a unique decomposition of a low rank matrix $\boldsymbol{Y} \in \mathbb{R}^{p\times n}$ that admits a sparse representation. Specifically, we consider $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X} \in \mathbb{R}^{p\times n}$ where matrix $\boldsymbol{A} \in \mathbb{R}^{p\times r}$ has full column rank, with $r < \min\{n, p\}$, and matrix $\boldsymbol{X} \in \mathbb{R}^{r\times n}$ is element-wise sparse. We prove that this low rank, sparse decomposition of $\boldsymbol{Y}$ can be uniquely identified, up to some intrinsic signed permutation. Our approach relies on solving a nonconvex optimization problem constrained over the unit sphere. Our geometric analysis for its nonconvex optimization landscape shows that any *strict* local solution is close to the ground truth solution, and can be recovered by a simple data-driven initialization followed with any second order descent algorithm. At last, we corroborate these theoretical results with numerical experiments.

1 Introduction

The problem of matrix decomposition has been a popular and fundamental topic under extensive investigations across several disciplines, including signal processing, machine learning, natural language processing [9, 10, 30, 44, 31, 7]. From the decomposition, one can construct efficient representation of the original data matrix. However, for any matrix $\boldsymbol{Y} \in \mathbb{R}^{p\times n}$ that can be factorized as a product of two matrices $\boldsymbol{A} \in \mathbb{R}^{p\times r}$ and $\boldsymbol{X} \in \mathbb{R}^{r\times n}$, there exist infinitely many decompositions, simply because one can use any $r \times r$ invertible matrix $\boldsymbol{Q}$ to construct $\boldsymbol{A}' = \boldsymbol{A}\boldsymbol{Q}$ and $\boldsymbol{X}' = \boldsymbol{Q}^{-1}\boldsymbol{X}$ such that $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X} = \boldsymbol{A}'\boldsymbol{X}'$, while $\boldsymbol{A}' \neq \boldsymbol{A}$ and $\boldsymbol{X}' \neq \boldsymbol{X}$. Thus, in various applications, additional structures and priors are being exploited to find a preferred representation [21, 14]. For example, principal component analysis (PCA) aims to find orthogonal representations which retain as much variations in $\boldsymbol{Y}$ as possible [16, 22], whereas independent component analysis (ICA) targets the representations of statistically independent non-Gaussian signals [25].

In this paper, we are interested in finding a unique *sparse* low-dimensional representation of $\boldsymbol{Y}$. To this end, we study the decomposition of a low rank matrix $\boldsymbol{Y} \in \mathbb{R}^{p\times n}$ that satisfies

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}, \tag{1.1}$$

where $\boldsymbol{A} \in \mathbb{R}^{p\times r}$ is an unknown deterministic matrix, with $r < \min\{n, p\}$, and $\boldsymbol{X} \in \mathbb{R}^{r\times n}$ is an unknown sparse matrix.

Formulation (1.1) is an important model problem in many applications. As columns of $\boldsymbol{Y}$ are viewed as linear combinations of columns of $\boldsymbol{A}$ with $\boldsymbol{X}$ being the sparse coefficient, (1.1) can be used to form overlapping clusters of the n columns of $\boldsymbol{Y}$ via the support of $\boldsymbol{X}$ with columns of $\boldsymbol{A}$ being viewed as r cluster centers [11, 6]. When we form a $p \times r$ low-dimensional representation of $\boldsymbol{Y}$ via sparse combinations, this greatly enhance the interpretability of the resulting representations [27, 20, 4], in the same spirit as the sparse PCA, but (1.1) generalizes to the factorization of non-orthogonal matrices.

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

To motivate our approach, we first consider the simple case that $\boldsymbol{A}$ has orthonormal columns, namely, $\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{I}_r$[1]. Then it is easy to see that the sparse coefficient matrix $\boldsymbol{X}$ is recovered by multiplying $\boldsymbol{Y}$ on the left by $\boldsymbol{A}^T$,

$$\boldsymbol{A}^T\boldsymbol{Y} = \boldsymbol{A}^T\boldsymbol{A}\boldsymbol{X} = \boldsymbol{X}. \tag{1.2}$$

The problem of finding such orthonormal matrix $\boldsymbol{A}$ boils down to successively finding a unit-norm direction $\boldsymbol{q}$ that renders $\boldsymbol{q}^T\boldsymbol{Y}$ as sparse as possible [33, 38, 34],

$$\min_{\boldsymbol{q}} \quad \left\|\boldsymbol{q}^T\boldsymbol{Y}\right\|_{\text{sparsity}} \qquad \text{s.t.} \quad \|\boldsymbol{q}\|_2 = 1. \tag{1.3}$$

However, the natural choice of sparsity penalty, either ℓ_0 or ℓ_1, leads to trivial and meaningless solutions, as there always exists $\boldsymbol{q}$ in the null space of $\boldsymbol{A}^T$ such that $\boldsymbol{q}^T\boldsymbol{Y} = \boldsymbol{0}$.

To avoid the null space of $\boldsymbol{A}^T$, we instead choose to find the unit direction $\boldsymbol{q}$ that maximizes the ℓ_4 norm of $\boldsymbol{q}^T\boldsymbol{Y}$ as

$$\max_{\boldsymbol{q}} \quad \left\|\boldsymbol{q}^T\boldsymbol{Y}\right\|_4 \qquad \text{s.t.} \quad \|\boldsymbol{q}\|_2 = 1. \tag{1.4}$$

The above formulation is based on the key observation that the objective value is maximized when $\boldsymbol{q}$ coincides with one column of $\boldsymbol{A}$ (see, Section 2, for details) while the objective value is zero when $\boldsymbol{q}$ lies in the null space of $\boldsymbol{A}^T$. The ℓ_4 norm objective function and its variants have been adopted as a sparsity regularizer in a line of recent works [29, 42, 41, 34, 40]. However, even with this new objective function, the null space of $\boldsymbol{A}^T$ persists as a challenge for solving the optimization problem: they form a flat region of local optimal solutions.

This paper characterizes the nonconvex optimization landscape of (1.4) and proposes a guaranteed procedure that avoids the flat region and provably recovers the global solution to (1.4), which corresponds to one column of $\boldsymbol{A}$. More specifically, we demonstrate that, despite the non-convexity, (1.4) still possesses benign geometric property in the sense that any *strict* local solution with *large* objective value is globally optimal and recovers one column of $\boldsymbol{A}$, up to its sign. See, Theorem 3.1 in Section 3.1 for the population level result and Theorem 3.4 for the finite sample result.

We further extend these results to the general case when $\boldsymbol{A}$ only has full column rank in Theorem 3.6 of Section 3.2. To recover a general $\boldsymbol{A}$ with full column rank, our procedure first resorts to a preconditioning procedure of $\boldsymbol{Y}$ proposed in Section 2.3 and then solves a optimization problem similar to (1.4). From our analysis of the optimization landscape, the intriguing problem boils down to developing algorithms to recover the nontrivial local solutions by avoiding regions with small objective values. We thus propose a simple initialization scheme in Section 4.1 and prove in Theorem 4.3 that such initialization, proceeded with any second order descent algorithm [19, 26], suffices to find the global solution, up to some statistical error. Our theoretical analysis provides the explicit convergence rate of the statistical error and characterizes its dependence on various dimensions, such as p, r and n, as well as the sparsity of $\boldsymbol{X}$.

Numerical simulation results[2] are provided in Section 5. Due to the space limitation, we defer all the proof along with our conclusions and discussion of several future directions of our work to Appendix.

Notations Throughout this paper, we use bold lowercase letters, like $\boldsymbol{a}$, to represent vectors and bold uppercase letters, like $\boldsymbol{A}$, to represent matrices. For matrix $\boldsymbol{X}$, $\boldsymbol{X}_{ij}$ denotes the entry at the i-th row and j-th column of $\boldsymbol{X}$, with $\boldsymbol{X}_{i\cdot}$ and $\boldsymbol{X}_{\cdot j}$ denoting the i-th row and j-th column of $\boldsymbol{X}$, respectively. Oftentimes, we write $\boldsymbol{X}_{\cdot j} = \boldsymbol{X}_j$ for simplicity. We use grad and Hess to represent the Riemannian gradient and Hessian. For any vector $\boldsymbol{v} \in \mathbb{R}^d$, we use $\|\boldsymbol{v}\|_q$ to denote its ℓ_q norm, for $1 \le q \le \infty$. The notation $\boldsymbol{v}^{\circ q}$ stands for $\{\boldsymbol{v}_i^q\}_i$. For matrices, we use $\|\cdot\|_F$ and $\|\cdot\|_{\text{op}}$ to denote the Frobenius norm and the operator norm, respectively. For any positive integer d, we write $[d] = \{1, 2, \ldots, d\}$. The unit sphere in d-dimensional real space

[1] $\boldsymbol{I}_r$ is the identity matrix of size $r \times r$.

[2] Codes to reproduce all experiment results are available at https://anonymous.4open.science/r/Unique_Fac_of_Low_Rank-99F5/

$\mathbb{R}^d$ is written as $\mathbb{S}^{d-1}$. For two sequences a_n and b_n, we write $a_n \lesssim b_n$ if there exists some constant $C > 0$ such that $a_n \leq C b_n$ for all n. Both uppercase C and lowercase c are reserved to represent numerical constants, whose values may vary line by line.

1.1 Related work

Finding the unique factorization of a matrix is an ill-posed problem in general due to infinitely many solutions. There exist several strands of studies from different contexts on finding the unique decomposition of $\boldsymbol{Y}$ by imposing additional structures on $\boldsymbol{A}$ and $\boldsymbol{X}$. We start by reviewing the literature which targets the sparse decomposition of $\boldsymbol{Y}$.

Dictionary learning The problems of dictionary learning (DL) [2, 37, 18, 38] and sparse blind deconvolution or convolutional dictionary learning [12, 28] study the unique decomposition of $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}$ where $\boldsymbol{X}$ is sparse and $\boldsymbol{A}$ has full row rank. In this case, the row space of $\boldsymbol{Y}$ lies in the row space of $\boldsymbol{X}$, suggesting to recover the sparse rows of $\boldsymbol{X}$ via solving the following problem,

$$\min_{\boldsymbol{q}} \quad \left\|\boldsymbol{q}^T \boldsymbol{Y}\right\|_1 \qquad \text{s. t.} \quad \boldsymbol{q} \neq 0. \tag{1.5}$$

Under certain scaling and incoherence conditions on $\boldsymbol{A}$, the objective achieves the minimum value when $\boldsymbol{q}$ is equal to one column of $\boldsymbol{A}$, at the same time $\boldsymbol{q}^T\boldsymbol{Y}$ recovers one sparse row of $\boldsymbol{X}$. This idea has been studied and modified in a strand of papers when $\boldsymbol{A}$ has full row rank [37, 38, 43, 29, 42, 34, 40, 36, 45]. In our context, the major difference rises in the matrix $\boldsymbol{A}$, which has full *column* rank rather than *row* rank, therefore minimizing $\left\|\boldsymbol{q}^T\boldsymbol{Y}\right\|_1$ as before only leads to some vector in the null space of $\boldsymbol{A}^T$, yielding the trivial zero objective value.

We would love to note that [34] uses the same objective function in (1.4) to study the problem of overcomplete dictionary learning (where $\boldsymbol{A}$ has full row rank), however the optimization landscape when $\boldsymbol{A}$ has full column rank is significantly different from that in the overcomplete setting. The more complicated optimization landscape in our setting brings additional difficulty of the analysis and requires a proper initialization in our proposed algorithm. We refer to Appendix B for detailed technical comparison with [34].

Sparse PCA Sparse principal component analysis (SPCA) is a popular method that recovers a unique decomposition of a low-rank matrix $\boldsymbol{Y}$ by utilizing the sparsity of its singular vectors. However, as being said, under $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}$, SPCA is only applicable when $\boldsymbol{X}$ coincides with the right singular vectors of $\boldsymbol{Y}$. Indeed, one formulation of SPCA is to solve

$$\max_{\boldsymbol{U} \in \mathbb{R}^{n \times r}} \quad \operatorname{tr}\left(\boldsymbol{U}^T \boldsymbol{Y}^T \boldsymbol{Y} \boldsymbol{U}\right) - \lambda \|\boldsymbol{U}\|_1, \qquad \text{s. t.} \quad \boldsymbol{U}^T \boldsymbol{U} = \boldsymbol{I}_r, \tag{1.6}$$

which is promising only if $\boldsymbol{X}$ corresponds to the right singular vectors of $\boldsymbol{Y}$. It is worth mentioning that among the various approaches of SPCA, the following one might be used to recover one sparse row of $\boldsymbol{X}$,

$$\min_{\boldsymbol{u}, \boldsymbol{v}} \quad \left\|\boldsymbol{Y} - \boldsymbol{u}\boldsymbol{v}^T\right\|_2^2 \; + \; \lambda \|\boldsymbol{v}\|_1 \qquad \text{s. t.} \quad \|\boldsymbol{u}\|_2 = 1. \tag{1.7}$$

This procedure was originally proposed by [47] and [35] together with an efficient algorithm by alternating the minimization between $\boldsymbol{u}$ and $\boldsymbol{v}$. However, there is no guarantee that the resulting solution recovers the ground truth.

Factor analysis Factor analysis is a popular statistical tool for constructing low-rank representations of $\boldsymbol{Y}$ by postulating $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{E}$ where $\boldsymbol{A} \in \mathbb{R}^{p \times r}$ is the so-called loading matrix with $r = \operatorname{rank}(\boldsymbol{A}) < \min\{n, p\}$, $\boldsymbol{X} \in \mathbb{R}^{r \times n}$ contains n realizations of a r-dimensional factor and $\boldsymbol{E}$ is some additive noise. Here only $\boldsymbol{Y}$ is observable. Factor analysis is mainly used to recover the low-dimension column space of $\boldsymbol{A}$ or the row space of $\boldsymbol{X}$, rather than to identify and recover the unique decomposition. Recently, [6] studied the unique decomposition of $\boldsymbol{Y}$ when the columns of $\boldsymbol{X}$ are i.i.d. realizations of a r-dimensional latent random factor. The unique decomposition is further used for (overlapping) clustering the rows of $\boldsymbol{Y}$ via the assignment matrix $\boldsymbol{A}$. To uniquely identify $\boldsymbol{A}$, [6] assumes that $\boldsymbol{A}$ contains at least one $r \times r$ identity matrix, coupled with other scaling conditions on $\boldsymbol{A}$ (we refer to [6] for detailed discussions of other existing conditions in the literature of factor models that ensure the unique decomposition of $\boldsymbol{Y}$ but require strong prior information on either $\boldsymbol{A}$ or $\boldsymbol{X}$). By contrast, we rely on the sparsity of $\boldsymbol{X}$ instead of $\boldsymbol{A}$ which is more general than requiring the existence of a $r \times r$ identity matrix.

NMF and topic models Such existence condition of identity matrix in either $\boldsymbol{A}$ or $\boldsymbol{X}$ has a variant in non-negative matrix factorization (NMF) [13] and topic models [3, 7, 8], also see the references therein, where $\boldsymbol{Y}$, $\boldsymbol{A}$ and $\boldsymbol{X}$ have non-negative entries. Since all $\boldsymbol{Y}$, $\boldsymbol{A}$ and $\boldsymbol{X}$ from model (1.1) are allowed to have arbitrary signs in our context, the approaches designed for NMF and topic models are inapplicable.

2 Formulation and Assumptions

The decomposition of $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}$ is not unique without further assumptions. To ensure the uniqueness of such decomposition, we rely on two assumptions on the matrices $\boldsymbol{A}$ and $\boldsymbol{X}$, stated in the following Section 2.1.

Our goal is to uniquely recover $\boldsymbol{A}$ from $\boldsymbol{Y}$, up to a some signed permutation. More precisely, we aim to recover columns of $\boldsymbol{A}\boldsymbol{P}$ for some signed permutation matrix $\boldsymbol{P} \in \mathbb{R}^{r\times r}$. To facilitate the understanding and motivate our approach, in Section 2.2 we first state our procedure for the unique recovery of $\boldsymbol{A}$ when $\boldsymbol{A}$ has orthonormal columns. Its theoretical analysis is presented in Section 3. Later in Section 2.3, we discuss how to extend our results to the case when $\boldsymbol{A}$ is a more general full column rank matrix under Assumption 2.2.

For now, we only focus on the recovery of one column of $\boldsymbol{A}$ as the remaining columns can be recovered via the same procedure after projecting $\boldsymbol{Y}$ onto the complement space spanned by the recovered columns of $\boldsymbol{A}$ (see Section D for detailed discussion).

2.1 Assumptions

We first resort to the matrix $\boldsymbol{X} \in \mathbb{R}^{r\times n}$ being element-wise sparse. The sparsity of $\boldsymbol{X}$ is modeled via the Bernoulli-Gaussian distribution, stated in the following assumption.

Assumption 2.1 *Assume $\boldsymbol{X}_{ij} = \boldsymbol{B}_{ij}\boldsymbol{Z}_{ij}$ for $i \in [r]$ and $j \in [n]$, where*

$$\boldsymbol{B}_{ij} \overset{i.i.d.}{\sim} Ber(\theta), \quad \boldsymbol{Z}_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2). \tag{2.1}$$

The Bernoulli-Gaussian distribution is popular for modeling sparse random matrices [37, 2, 1, 38]. The overall sparsity level of $\boldsymbol{X}$ is controlled by θ, the parameter of the Bernoulli distribution. We remark that the Gaussianity is assumed only to simplify the proof and to obtain more transparent deviation inequalities between quantities related with $\boldsymbol{X}$ and their population counterparts. Both our approach and analysis can be generalized to cases where $\boldsymbol{Z}_{ij}$ are centered i.i.d. sub-Gaussian random variables.

We also need another condition on the matrix $\boldsymbol{A}$. To see this, note that even when $\boldsymbol{A}$ were known, recovering $\boldsymbol{X}$ from $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}$ requires $\boldsymbol{A}$ to have full column rank. We state this in the following assumption.

Assumption 2.2 *Assume the matrix $\boldsymbol{A} \in \mathbb{R}^{p\times r}$ has $\operatorname{rank}(\boldsymbol{A}) = r$ with $\|\boldsymbol{A}\|_{op} = 1$.*

The unit operator norm of $\boldsymbol{A}$ is assumed without loss of generality as one can always re-scale σ^2, the variance of $\boldsymbol{X}$, by $\|\boldsymbol{A}\|_{\mathrm{op}}$.

2.2 Recovery of the orthonormal columns of A

In this section, we consider the recovery of one column of $\boldsymbol{A}$ when $\boldsymbol{A}$ is a semi-orthogonal matrix satisfying the following assumption.

Assumption 2.3 *Assume $\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{I}_r$.*

Our approach recovers columns of $\boldsymbol{A}$ one at a time by adopting the ℓ_4 maximization to penalize the sparsity of rows of matrix $\boldsymbol{X}$. Its rationale is based on the following lemma, assuming the orthogonality among columns of $\boldsymbol{A}$.

Lemma 2.4 *Under Assumption 2.3, solving the following problem*

$$\max_{\boldsymbol{q}} \quad \left\|\boldsymbol{A}^T\boldsymbol{q}\right\|_4^4 \quad \text{s.t.} \quad \|\boldsymbol{q}\|_2 = 1 \tag{2.2}$$

recovers one column of $\boldsymbol{A}$, up to its sign.

Intuitively, under Assumption 2.3, we have $\|\boldsymbol{A}^T\boldsymbol{q}\|_2 \leq 1$ for any unit vector $\boldsymbol{q}$. Therefore, criterion (2.2) seeks a vector $\boldsymbol{A}^T\boldsymbol{q}$ within the unit ball to maximize its ℓ_4 norm. When $\boldsymbol{q}$ corresponds to one column of $\boldsymbol{A}$, that is, $\boldsymbol{q} = \boldsymbol{a}_i$ for any $i \in [r]$, we have the largest objective $\|\boldsymbol{A}^T\boldsymbol{a}_i\|_4^4 = 1$. This ℓ_4 norm maximization approach has been used in several related literature, for instance, sparse blind deconvolution [42, 29], complete and over-complete dictionary

learning [41, 40, 34], independent component analysis [24, 23] and tensor decomposition [17].

The appealing property of maximizing the ℓ_4 norm is its benign geometry landscape under the unit sphere constraint. Indeed, despite of the non-convexity of (2.2), our result in Theorem 3.1 implies that any strict location solution to (2.2) is globally optimal. This enables us to use any second order gradient accent method to solve (2.2).
Motivated by Lemma 2.4, since we only have access to $\boldsymbol{Y} \in \mathbb{R}^{p\times n}$, we propose to solve the following problem to recover one column of $\boldsymbol{A}$,

$$\min_{\boldsymbol{q}} (\boldsymbol{q}) \doteq -\frac{1}{12\theta\sigma^4 n} \left\|\boldsymbol{Y}^T\boldsymbol{q}\right\|_4^4 \qquad \text{s.t.} \quad \|\boldsymbol{q}\|_2 = 1. \tag{2.3}$$

The scalar $(12\theta\sigma^4 n)^{-1}$ is a normalization constant. The following lemma justifies the usage of (2.3) and also highlights the role of the sparsity of $\boldsymbol{X}$.

Lemma 2.5 *Under model (1.1) and Assumption 2.1, we have*

$$\mathbb{E}\left[F(\boldsymbol{q})\right] = -\frac{1}{4}\left[(1-\theta)\left\|\boldsymbol{A}^T\boldsymbol{q}\right\|_4^4 + \theta\left\|\boldsymbol{A}^T\boldsymbol{q}\right\|_2^4\right] \tag{2.4}$$

where the expectation is taken over the randomness of $\boldsymbol{X}$.

Remark 2.6 (Role of the sparsity parameter θ) *Lemma 2.5 implies that, for large n, solving (2.3) approximately finds the solution to*

$$\min_{\boldsymbol{q}} (\boldsymbol{q}) \doteq -\frac{1}{4}\left[(1-\theta)\left\|\boldsymbol{A}^T\boldsymbol{q}\right\|_4^4 + \theta\left\|\boldsymbol{A}^T\boldsymbol{q}\right\|_2^4\right] \qquad \text{s.t.} \quad \|\boldsymbol{q}\|_2 = 1 \tag{2.5}$$

The objective function is a convex combination of $\|\boldsymbol{A}^T\boldsymbol{q}\|_4^4$ and $\|\boldsymbol{A}^T\boldsymbol{q}\|_2^2$ with coefficients depending on the magnitude of θ. In view of Lemma 2.4, it is easy to see that solving (2.5) recovers one column of $\boldsymbol{A}$, up to the sign, as long as $\theta < 1$. However, the magnitude of θ controls the benignness of the geometry landscape of (2.5). When θ is small, or $\boldsymbol{X}$ is sufficiently sparse, we essentially solve (2.2) which has the most benign landscape. On the other hand, when $\theta \to 1$, the landscape of (2.5) is mostly determined by the eigenvalue problem[3] which maximizes $\|\boldsymbol{A}^T\boldsymbol{q}\|_2$ subject to $\|\boldsymbol{q}\|_2 = 1$. We will demonstrate that when $\boldsymbol{X}$ is sufficiently sparse, second order descent algorithm with a simple initialization finds the globally optimal solution to (2.3) in Section 3.

2.3 Recovery of the non-orthogonal columns of A

In this section, we discuss how to extend our procedure to recover $\boldsymbol{A}$ from $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}$ when $\boldsymbol{A}$ is a general full column rank matrix satisfying Assumption 2.2. The main idea is to first resort to a preconditioning procedure of $\boldsymbol{Y}$ such that the preconditioned $\boldsymbol{Y}$ has the decomposition $\bar{\boldsymbol{A}}\bar{\boldsymbol{X}}$, up to some small perturbation, where $\bar{\boldsymbol{A}}$ satisfies Assumption 2.3 and $\bar{\boldsymbol{X}}$ satisfies Assumption 2.1 with $\sigma^2 = 1$. Then we apply our procedure in Section 2.2 to recover $\bar{\boldsymbol{A}}$. The recovered $\bar{\boldsymbol{A}}$ is further used to recover the original $\boldsymbol{A}$.

To precondition $\boldsymbol{Y}$, we propose to left multiply $\boldsymbol{Y}$ by the following matrix

$$\boldsymbol{D} \doteq \left[\left(\boldsymbol{Y}\boldsymbol{Y}^T\right)^+\right]^{1/2} \in \mathbb{R}^{p\times p} \tag{2.6}$$

where $\boldsymbol{M}^+$ denotes the Moore-Penrose inverse of any matrix $\boldsymbol{M}$. The resulting preconditioned $\boldsymbol{Y}$ satisfies

$$\bar{\boldsymbol{Y}} \doteq \boldsymbol{D}\boldsymbol{Y} = \bar{\boldsymbol{A}}\bar{\boldsymbol{X}} + \boldsymbol{E} \tag{2.7}$$

with $\bar{\boldsymbol{A}}$ satisfying Assumption 2.3, $\bar{\boldsymbol{X}} = \boldsymbol{X}/\sqrt{\theta n\sigma^2}$ and $\boldsymbol{E}$ being a perturbation matrix with small entries. We refer to Proposition 3.5 below for its precise statements.

Analogous to (2.3), we propose to recover one column of $\bar{\boldsymbol{A}}$ by solving the following problem

$$\min_{\|\boldsymbol{q}\|_2=1} \quad F_{\mathrm{g}}(\boldsymbol{q}) \doteq -\frac{\theta n}{12}\left\|\bar{\boldsymbol{Y}}^T\boldsymbol{q}\right\|_4^4 \tag{2.8}$$

Theoretical guarantees of this procedure are provided in Section 3.2. After recovering one column of $\bar{\boldsymbol{A}}$, the remaining columns of $\bar{\boldsymbol{A}}$ can be successively recovered via the procedure in Section D. In the end, $\boldsymbol{A}$ can be recovered by first inverting the preconditioning matrix $\boldsymbol{D}$ as $\boldsymbol{D}^{-1}\bar{\boldsymbol{A}}$ and then re-scaling its largest singular value to 1.

[3] When $\boldsymbol{A}$ is orthonormal, this eigenvalue problem processes the worst landscape as there are infinitely many solutions obtaining the same eigenvalue.

3 Theoretical Guarantees

We provide theoretical guarantees for our procedure (2.3) in Section 3.1 when $\boldsymbol{A}$ has orthonormal columns. The theoretical guarantees of (2.8) for recovering a general full column rank $\boldsymbol{A}$ are stated in Section 3.2.

3.1 Theoretical guarantees for semi-orthonormal A

In this section, we provide guarantees for our procedure by characterizing the solution to (2.3) when $\boldsymbol{A}$ satisfies Assumption 2.3.

As the objective function $F(\boldsymbol{q})$ in (2.3) concentrates around $f(\boldsymbol{q})$ in (2.5), it is informative to first analyze the solution to (2.5). Although (2.5) is a nonconvex problem and has multiple local solutions, Theorem 3.1 below guarantees that any strict local solution to (2.5) is globally optimal, in the sense that, it recovers one column of $\boldsymbol{A}$, up to its sign. We introduce the null region R_0 of our objective in (2.5),

$$R_0 = \left\{\boldsymbol{q} \in \mathbb{S}^{p-1} : \|\boldsymbol{A}^T\boldsymbol{q}\|_\infty = 0\right\}. \tag{3.1}$$

Theorem 3.1 (Population case) *Under Assumption 2.3, assume $\theta \leq 1/6$. Any local solution $\bar{\boldsymbol{q}}$ to (2.5), that is not in R_0, satisfies*

$$\bar{\boldsymbol{q}} = \boldsymbol{A}\boldsymbol{P}_{\cdot 1}$$

for some signed permutation matrix $\boldsymbol{P} \in \mathbb{R}^{r\times r}$.

The detailed proof of Theorem 3.1 is deferred to Appendix F.3. We only offer an outline of our analysis below.

The proof of Theorem 3.1 relies on analysis of the optimization landscape of (2.5) on disjoint partitions of $\mathbb{S}^{p-1} = \{\boldsymbol{q} \in \mathbb{R}^p : \|\boldsymbol{q}\|_2 = 1\}$, defined as

$$R_1 \doteq R_1(C_\star) = \left\{\boldsymbol{q} \in \mathbb{S}^{p-1} : \|\boldsymbol{A}^T\boldsymbol{q}\|_\infty^2 \geq C_\star\right\}, \quad R_2 = \mathbb{S}^{p-1} \setminus (R_0 \cup R_1). \tag{3.2}$$

Here $C_\star$ is any fixed constant between 0 and 1. The upper bound follows from the inequality that $\|\boldsymbol{A}^T\boldsymbol{q}\|_\infty = \max_k |\boldsymbol{a}_k^T\boldsymbol{q}| \leq \|\boldsymbol{a}_k\|_2\|\boldsymbol{q}\|_2 = 1$ for any $\boldsymbol{q} \in \mathbb{S}^{p-1}$. The region R_0 can be easily avoided by choosing the initialization such that the objective function $f(\boldsymbol{q})$ is not equal to zero. For R_1 and R_2, we are able to show the following results. Let $\operatorname{Hess} f(\boldsymbol{q})$ be the Riemannian Hessian matrix of (2.5) at any point $\boldsymbol{q} \in \mathbb{S}^{p-1}$.

(1) Optimization landscape for R_1:

Lemma 3.2 *Assume $\theta < 1$. Any local solution $\bar{\boldsymbol{q}} \in R_1(C_\star)$ to (2.5) with*

$$C_\star > \frac{1}{2}\sqrt{\frac{\theta}{1-\theta}} \tag{3.3}$$

recovers one column of $\boldsymbol{A}$, that is,

$$\bar{\boldsymbol{q}} = \boldsymbol{A}\boldsymbol{P}_{\cdot 1}$$

for some signed permutation matrix $\boldsymbol{P}$.

Lemma 3.2 shows that any critical point $\bar{\boldsymbol{q}} \in R_1$ is either a strict saddle point that there exists a direction along which the Hessian is negative, or satisfies the second order optimality condition and is equal to one column of $\boldsymbol{A}$, up to its sign.

(2) Optimization landscape for R_2:

Lemma 3.3 *Assume $\theta < 1/3$. For any point $\boldsymbol{q} \in R_2(C_\star)$ with*

$$C_\star \leq \frac{1-3\theta}{2},$$

there exists $\boldsymbol{v}$ such that

$$\boldsymbol{v}^T \operatorname{Hess} f(\boldsymbol{q})\, \boldsymbol{v} < 0.$$

Lemma 3.3 implies that any critical point in R_2 is a saddle point that can be escaped by negative curvature. Hence there is no local solution to (2.5) in the region R_2.

Theorem 3.1 thus follows from Lemma 3.2 and Lemma 3.3, provided that

$$\sqrt{\frac{\theta}{1-\theta}} < 1 - 3\theta. \tag{3.4}$$

Condition (3.4) puts restrictions on the upper bound of θ. It is easy to see that (3.4) holds for any $\theta \leq 1/6$. As discussed in Remark 2.6, a smaller θ leads to a more benign optimization landscape.

In light of Theorem 3.1, we now provide guarantees for the solution to the finite sample problem (2.3) in the following theorem. Define the sample analogue of the null region R_0 in (3.1) as

$$R_0'(c_\star) \;\doteq\; \left\{\boldsymbol{q} \in \mathbb{S}^{p-1} : \|\boldsymbol{A}^T\boldsymbol{q}\|_\infty^2 \;\leq\; c_\star\right\} \tag{3.5}$$

for any given value $c_\star \in [0, 1)$.

Theorem 3.4 (Finite sample case) *Under Assumptions 2.1 and 2.3, assume $\theta \in (0, 1/9]$ and*

$$n \geq C \max\left\{\frac{r^2}{c_\star},\; \log^2 n\right\} \frac{r \log n}{\theta c_\star} \tag{3.6}$$

for some sufficiently large constant $C > 0$ and any $c_\star \in (0, 1/4]$. Then with probability at least $1 - cn^{-c'}$, any local solution $\bar{\boldsymbol{q}}$ to (2.3) that is not in $R_0'(c_\star)$ satisfies

$$\|\bar{\boldsymbol{q}} - \boldsymbol{A}\boldsymbol{P}_{\cdot 1}\|_2^2 \;\lesssim\; \sqrt{\frac{r^2 \log n}{\theta n}} + \left(\theta r^2 + \frac{\log^2 n}{\theta}\right) \frac{r \log n}{n} \tag{3.7}$$

for some signed permutation matrix $\boldsymbol{P}$.

Here we defer our discussion of technical details and full proof in section C.

3.2 Theoretical guarantees for general full column rank A

In this section, we provide theoretical guarantees for our procedure of recovering a general full column rank matrix $\boldsymbol{A}$ under Assumption 2.2.

Recall from Section 2.3 that our approach first preconditions $\boldsymbol{Y}$ by using $\boldsymbol{D}$ from (2.6). The following proposition provides guarantees for the preconditioned $\boldsymbol{Y}$, denoted as $\bar{\boldsymbol{Y}} = \boldsymbol{D}\boldsymbol{Y}$. The proof is deferred to Appendix F.5. Write the SVD of $\boldsymbol{A} = \boldsymbol{U}_A \boldsymbol{D}_A \boldsymbol{V}_A^T$ with $\boldsymbol{U}_A \in \mathbb{R}^{p \times r}$ and $\boldsymbol{V}_A \in \mathbb{R}^{r \times r}$ being, respectively, the left and right singular vectors.

Proposition 3.5 *Under Assumptions 2.1 and 2.2, assume $n \geq Cr/\theta^2$ for some sufficiently large constant $C > 0$. With probability greater than $1 - 2e^{-c'r}$, one has*

$$\bar{\boldsymbol{Y}} = \bar{\boldsymbol{A}}\bar{\boldsymbol{X}} + \boldsymbol{E} \tag{3.8}$$

where $\bar{\boldsymbol{A}} = \boldsymbol{U}_A \boldsymbol{V}_A^T$, $\bar{\boldsymbol{X}} = \boldsymbol{X}/\sqrt{\theta n \sigma^2}$ and $\boldsymbol{E} = \bar{\boldsymbol{A}}\boldsymbol{\Delta}\bar{\boldsymbol{X}}$ with

$$\|\boldsymbol{\Delta}\|_{op} \;\leq\; c'' \frac{1}{\theta}\sqrt{\frac{r}{n}}. \tag{3.9}$$

Here c' and c'' are positive constants.

Proposition 3.5 implies that, when $n \geq Cr/\theta^2$, the preconditioned $\boldsymbol{Y}$ satisfies

$$\bar{\boldsymbol{Y}} = \bar{\boldsymbol{A}}(\boldsymbol{I}_r + \boldsymbol{\Delta})\bar{\boldsymbol{X}} \approx \bar{\boldsymbol{A}}\bar{\boldsymbol{X}} \tag{3.10}$$

with $\bar{\boldsymbol{A}}^T\bar{\boldsymbol{A}} = \boldsymbol{I}_r$. This naturally leads us to apply our procedure in Section 2.2 to recover columns of $\bar{\boldsymbol{A}}$ via (2.8). We formally show in Theorem 3.6 below that any local solution to (2.8) approximates recover one column of $\bar{\boldsymbol{A}}$ up to a signed permutation matrix. Similar to (3.5), define

$$R_0''(c_\star) \;\doteq\; \left\{\boldsymbol{q} \in \mathbb{S}^{p-1} : \left\|\bar{\boldsymbol{A}}^T\boldsymbol{q}\right\|_\infty^2 \;\leq\; c_\star\right\} \tag{3.11}$$

for some given value $c_\star \in [0, 1)$.

Theorem 3.6 *Under Assumption 2.1 and 2.2, assume $\theta \in (0, 1/9]$ and*

$$n \geq C \frac{r}{c_\star \theta} \max\left\{\log^3 n,\; \frac{\log n}{c_\star \theta \sqrt{\theta}},\; \frac{\log^2 n}{c_\star \theta},\; \frac{r}{c_\star \sqrt{\theta}},\; \frac{r^2 \log n}{c_\star}\right\}. \tag{3.12}$$

Then with probability at least $1 - cn^{-c'} - 4e^{-c''r}$, any solution $\bar{\boldsymbol{q}}$ to (2.8) that is not in Region $R_0''(c_\star)$ satisfies

$$\left\|\bar{\boldsymbol{q}} - \bar{\boldsymbol{A}}\boldsymbol{P}_{\cdot 1}\right\|_2^2 \;\lesssim\; \sqrt{\frac{r \log n}{\theta^2 n}} + \sqrt{\frac{r^2 \log n}{\theta n}} + \left(\theta r^2 + \frac{\log^2 n}{\theta}\right) \frac{r \log n}{n} \tag{3.13}$$

for some signed permutation matrix $\boldsymbol{P}$.

The proof of Theorem 3.6 can be found in Appendix F.6. Due to the preconditioning step, the requirement of the sample size in (3.12) is slightly stronger than (3.6), whereas the estimation error of $\bar{\boldsymbol{q}}$ only has an additional $\sqrt{r \log n/(\theta^2 n)}$ term comparing to (3.7).

Theorem 3.6 requires to avoid the null region $R_0''(c_\star)$ in (3.11). We provide a simple initialization in the next section that provably avoids $R_0^{''}$. Furthermore, every iterate of any descent algorithm based on such initialization is provably not in $R_0^{''}$ either.

4 Complete Algorithm and Provable Recovery

In this section, we present a complete pipeline for recovering $\boldsymbol{A}$ from $\boldsymbol{Y}$. So far we have established that every local solution to (2.8), that is not in $R_0''(c_\star)$, approximately recovers one column of $\bar{\boldsymbol{A}} = \boldsymbol{U}_A \boldsymbol{V}_A^T$. To our end, we will discuss: (1) a data-driven initialization in Section 4.1 which, together with Theorem 3.6, provably recovers one column of $\bar{\boldsymbol{A}}$; (2) a deflation procedure [37, 38, 34] in Section D that sequentially recovers all remaining columns of $\bar{\boldsymbol{A}}$. Due to the limitation of space we defer our discussion of deflation procedure in appendix.

4.1 Initialization

Our goal is to provide a simple initialization such that solving (2.8) via any second order descent algorithm provably recovers one column of $\bar{\boldsymbol{A}}$. According to Theorem 3.6, such an initialization needs to guarantee the following conditions.

- **Condition I:** The initial point $\boldsymbol{q}^{(0)}$ does not fall into region $R_0^{''}(c_\star)$ for some $c_\star$ satisfying (3.12) in Theorem 3.6.
- **Condition II:** The updated iterates $\boldsymbol{q}^{(k)}$, for all $k \geq 1$, stay away from $R_0''(c_\star)$ as well.

We propose the following initialization

$$\boldsymbol{q}^{(0)} = \frac{\bar{\boldsymbol{Y}} \boldsymbol{1}_n}{\|\bar{\boldsymbol{Y}} \boldsymbol{1}_n\|_2} \in \mathbb{S}^{p-1}. \tag{4.1}$$

The following two lemmas guarantee that both **Condition I** and **Condition II** are met for this choice. Their proofs can be found in Appendices F.7 and F.8.

Lemma 4.1 *Under Assumption 2.1 and 2.2, assume $\theta \in (0, 1/9]$ and*

$$n \geq C \frac{r^2}{\theta} \max\left\{ \log^3 n,\ \frac{r \log n}{\theta \sqrt{\theta}},\ \frac{r \log^2 n}{\theta},\ \frac{r^2}{\sqrt{\theta}},\ r^3 \log n \right\}. \tag{4.2}$$

holds, then, with probability at least $1 - 2e^{-cr}$, the initialization $\boldsymbol{q}^{(0)}$ in (4.1) is not in region $R_0^{''}(c_\star)$ with $c_\star = 1/(2r)$.

Lemma 4.2 *Let $\boldsymbol{q}^{(k)}$, for $k \geq 1$, be any updated iterate from solving (2.3) by using any descent algorithm with the initial point $\boldsymbol{q}^{(0)}$ chosen as (4.1). If*

$$n \geq C \frac{r^2}{\theta} \max\left\{ \log^3 n,\ \frac{r \log n}{\theta \sqrt{\theta}},\ \frac{r^2}{\sqrt{\theta}},\ \theta^2 r^2 \log n \right\} \tag{4.3}$$

holds, then, with probability at least $1 - cn^{-c'} - 2e^{-c''r}$, one has

$$\boldsymbol{q}^{(k)} \notin R_0''(c_\star), \qquad \text{for all } k \geq 1,$$

with $c_\star = 1/(2r)$.

Combining Lemmas 4.1 and 4.2 together with Theorem 3.6 readily yields the following theorem.

Theorem 4.3 *Under Assumptions 2.1 and 2.2, assume $\theta \in (0, 1/9]$ and (4.2) holds. Let $\bar{\boldsymbol{q}}$ be any local solution to (2.8) from any second order descent algorithm with the initial point chosen as (4.1). With probability at least $1 - cn^{-c'} - 4e^{-c''r}$, one has*

$$\|\bar{\boldsymbol{q}} - \bar{\boldsymbol{A}} \boldsymbol{P}_{\cdot 1}\|_2^2 \lesssim \sqrt{\frac{r \log n}{\theta^2 n}} + \sqrt{\frac{r^2 \log n}{\theta n}} + \frac{r \log^3 n}{\theta n}$$

for some signed permutation matrix $\boldsymbol{P}$.

Theorem 4.3 provides the guarantees for using any second order descent algorithms [32, 5] to solve (2.8) with the initialization chosen in (4.1).

5 Experiments

In this section we verify the empirical performance of our proposed procedure for recovering $\boldsymbol{A}$ under model (1.1) in different scenarios. Due to the space limit, we defer more experiments to the Appendix of this paper.

Experiment setup

To generate the data $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}$, we generate the columns of $\boldsymbol{A}$ by using the normalized left singular vectors of $\boldsymbol{R} \in \mathbb{R}^{p\times r}$ where $\boldsymbol{R}_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$. The sparse coefficient matrix $\boldsymbol{X} \in \mathbb{R}^{r\times n}$ are generated as $\boldsymbol{X}_{ij} \overset{i.i.d.}{\sim} \mathrm{BG}(\theta)$. To evaluate the success of recovering one column vector of $\boldsymbol{A}$ for any estimate $\boldsymbol{q} \in \mathbb{S}^{p-1}$, we use the following criterion,

$$\mathbf{Err}\left(\boldsymbol{q}\right) = \min_{1\le i\le r} \left(1 - \left|\langle \boldsymbol{q}, \boldsymbol{a}_i\rangle\right|\right) \tag{5.1}$$

If $\mathbf{Err}\left(\boldsymbol{q}\right) \le \rho_e$, we say the vector $\boldsymbol{q}$ recovers the ground-truth column vector of $\boldsymbol{A}$. We choose $\rho_e = 1\times 10^{-2}$ in our simulation settings. To evaluate the recovery of the whole matrix $\boldsymbol{A}$, we use the following normalized Frobenius norm between any estimate $\boldsymbol{A}_{est}$ and the true $\boldsymbol{A}$:

$$\min_{\boldsymbol{P}} \quad \frac{1}{\sqrt{r}} \left\|\boldsymbol{A}_{est} - \boldsymbol{A}\boldsymbol{P}\right\|_F \quad \text{s.t.} \quad \boldsymbol{P} \text{ is a signed permutation matrix.} \tag{5.2}$$

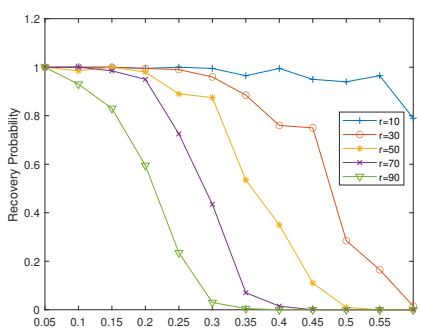
We first evaluate the probability of successfully recovering one column of $\boldsymbol{A}$ in two scenarios. In the first case, we vary simultaneously θ and r while in the second case we change n and r. We then evaluate the performance of our procedure, Algorithm 0 in Section D, for recovering the full matrix $\boldsymbol{A}$.

Recovery probability with varying θ and r

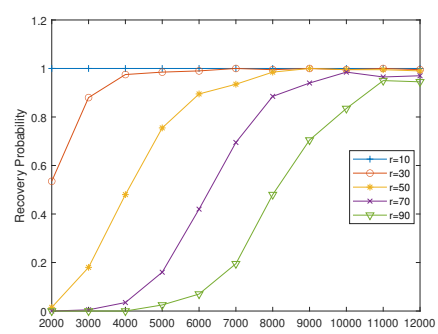
We fix $p = 100$ and $n = 1.2\times 10^4$ while vary $\theta \in \{0.05, 0.1, \ldots, 0.6\}$ and $r \in \{10, 30, \ldots, 90\}$. For each pair of (θ, r), we repeatedly generate 200 data sets and apply our procedure. The averaged recovery probability of our procedure over the 200 replicates is shown in Figure 1a. The recovery probability gets larger as either θ or r decreases, in line with Theorem 3.4.

Recovery probability with varying n and r

Here we fix $p = 100$ and set the sparsity parameter $\theta = 0.1$. We vary $r \in \{10, 30, \ldots, 90\}$ and $n \in \{2000, 3000, \ldots, 12000\}$. Figure 1b shows the averaged recovery probability of our procedure over 200 replicates in each setting. Our procedure performs increasingly better as n increases, as expected from Theorem 3.4.



(a) Recovery probability versus θ: the averaged probability of successful recovery for different θ and r with $p = 100$ and $n = 1.2\times 10^4$.



(b) Recovery probability versus n: the averaged probability of successful recovery for different n and r with $p = 100$ and $\theta = 0.1$.

References

[1] Alekh Agarwal, Animashree Anandkumar, Prateek Jain, and Praneeth Netrapalli. Learning sparsely used overcomplete dictionaries via alternating minimization. *SIAM Journal on Optimization*, 26(4):2775–2799, 2016.

[2] Alekh Agarwal, Animashree Anandkumar, and Praneeth Netrapalli. Exact recovery of sparsely used overcomplete dictionaries. *stat*, 1050:8–39, 2013.

[3] Sanjeev Arora, Rong Ge, Yonatan Halpern, David Mimno, Ankur Moitra, David Sontag, Yichen Wu, and Michael Zhu. A practical algorithm for topic modeling with provable guarantees. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 280–288. PMLR, 17–19 Jun 2013.

[4] Tom Baden, Philipp Berens, Katrin Franke, Miroslav Román Rosón, Matthias Bethge, and Thomas Euler. The functional diversity of retinal ganglion cells in the mouse. *Nature*, 529(7586):345–350, 2016.

[5] Roberto Battiti. First-and second-order methods for learning: between steepest descent and newton's method. *Neural computation*, 4(2):141–166, 1992.

[6] Xin Bing, Florentina Bunea, Yang Ning, and Marten Wegkamp. Adaptive estimation in structured factor models with applications to overlapping clustering. *Ann. Statist.*, 48(4):2055–2081, 08 2020.

[7] Xin Bing, Florentina Bunea, and Marten Wegkamp. A fast algorithm with minimax optimal guarantees for topic models with an unknown number of topics. *Bernoulli*, 26(3):1765–1796, 08 2020.

[8] Xin Bing, Florentina Bunea, and Marten Wegkamp. Optimal estimation of sparse topic models. *Journal of Machine Learning Research*, 21(177):1–45, 2020.

[9] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.

[10] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):1–37, 2011.

[11] Guanhua Chen, Patrick F. Sullivan, and Michael R. Kosorok. Biclustering with heterogeneous variance. *Proceedings of the National Academy of Sciences*, 110(30):12253–12258, 2013.

[12] Sky C Cheung, John Y Shin, Yenson Lau, Zhengyu Chen, Ju Sun, Yuqian Zhang, Marvin A Müller, Ilya M Eremin, John N Wright, and Abhay N Pasupathy. Dictionary learning in fourier-transform scanning tunneling spectroscopy. *Nature communications*, 11(1):1–11, 2020.

[13] David Donoho and Victoria Stodden. When does non-negative matrix factorization give a correct decomposition into parts? Advances in Neural Information Processing Systems, January 2004.

[14] Jicong Fan, Yuqian Zhang, and Madeleine Udell. Polynomial matrix completion for missing data imputation and transductive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3842–3849, 2020.

[15] Simon Foucart and Holger Rauhut. *A Mathematical Introduction to Compressive Sensing*. Birkhäuser Basel, 2013.

[16] Karl Pearson F.R.S. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[17] Rong Ge and Tengyu Ma. On the optimization landscape of tensor decompositions. *Mathematical Programming*, pages 1–47, 2020.

[18] Quan Geng and John Wright. On the local correctness of 1-minimization for dictionary learning. In *2014 IEEE International Symposium on Information Theory*, pages 3180–3184. IEEE, 2014.

[19] Donald Goldfarb. Curvilinear path steplength algorithms for minimization which use directions of negative curvature. *Mathematical programming*, 18(1):31–40, 1980.

[20] Kelly Gravuer, Jon J. Sullivan, Peter A. Williams, and Richard P. Duncan. Strong human association with plant invasion success for trifolium introductions to new zealand. *Proceedings of the National Academy of Sciences*, 105(17):6344–6349, 2008.

[21] Benjamin Haeffele, Eric Young, and Rene Vidal. Structured low-rank matrix factorization: Optimality, algorithm, and applications to image processing. In *International conference on machine learning*, pages 2007–2015. PMLR, 2014.

[22] Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

[23] Aapo Hyvarinen. A family of fixed-point algorithms for independent component analysis. In *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 3917–3920. IEEE, 1997.

[24] Aapo Hyvärinen and Erkki Oja. A fast fixed-point algorithm for independent component analysis. *Neural computation*, 9(7):1483–1492, 1997.

[25] Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.

[26] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *International Conference on Machine Learning*, pages 1724–1732. PMLR, 2017.

[27] Ian T. Jolliffe. Rotation of principal components: Some comments. *Journal of Climatology*, 7(5):507–510, 1987.

[28] Han-Wen Kuo, Yenson Lau, Yuqian Zhang, and John Wright. Geometry and symmetry in short-and-sparse deconvolution. In *International Conference on Machine Learning*, pages 3570–3580. PMLR, 2019.

[29] Yanjun Li and Yoram Bresler. Global geometry of multichannel sparse blind deconvolution on the sphere. *CoRR*, abs/1805.10437, 2018.

[30] Zhouchen Lin. A review on low-rank models in data analysis. *Big Data & Information Analytics*, 1(2&3):139, 2016.

[31] Cun Mu, Yuqian Zhang, John Wright, and Donald Goldfarb. Scalable robust matrix recovery: Frank–wolfe meets proximal methods. *SIAM Journal on Scientific Computing*, 38(5):A3291–A3317, 2016.

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

[33] Qing Qu, Ju Sun, and John Wright. Finding a sparse vector in a subspace: Linear sparsity using alternating directions. In *Advances in Neural Information Processing Systems*, pages 3401–3409, 2014.

[34] Qing Qu, Yuexiang Zhai, Xiao Li, Yuqian Zhang, and Zhihui Zhu. Analysis of the optimization landscapes for overcomplete representation learning. *arXiv preprint arXiv:1912.02427*, 2019.

[35] Haipeng Shen and Jianhua Z. Huang. Sparse principal component analysis via regularized low rank matrix approximation. *Journal of Multivariate Analysis*, 99(6):1015 – 1034, 2008.

[36] Laixi Shi and Yuejie Chi. Manifold gradient descent solves multi-channel sparse blind deconvolution provably and efficiently. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5730–5734. IEEE, 2020.

[37] Daniel A. Spielman, Huan Wang, and John Wright. Exact recovery of sparsely-used dictionaries. In *Conference on Learning Theory*, 2012.

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

[39] Roman Vershynin. *Introduction to the non-asymptotic analysis of random matrices*, page 210–268. Cambridge University Press, 2012.

[40] Yuexiang Zhai, Hermish Mehta, Zhengyuan Zhou, and Yi Ma. Understanding l4-based dictionary learning: Interpretation, stability, and robustness. In *International Conference on Learning Representations*, 2020.

[41] Yuexiang Zhai, Zitong Yang, Zhenyu Liao, John Wright, and Yi Ma. Complete dictionary learning via l4-norm maximization over the orthogonal group. *J. Mach. Learn. Res.*, 21:165:1–165:68, 2020.

[42] Yuqian Zhang, Han-Wen Kuo, and John Wright. Structured local optima in sparse blind deconvolution. *IEEE Transactions on Information Theory*, 66(1):419–452, 2020.

[43] Yuqian Zhang, Yenson Lau, Han-Wen Kuo, Sky Cheung, Abhay Pasupathy, and John Wright. On the global geometry of sphere-constrained sparse blind deconvolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[44] Yuqian Zhang, Cun Mu, Han-Wen Kuo, and John Wright. Toward guaranteed illumination models for non-convex objects. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 937–944, 2013.

[45] Yuqian Zhang, Qing Qu, and John Wright. From symmetry to geometry: Tractable nonconvex problems. *arXiv preprint arXiv:2007.06753*, 2020.

[46] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.

[47] Hui Zou, Trevor Hastie, and Robert Tibshirani. Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286, 2006.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] Please see our abstract and introduction 1
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] Please check section 2 and 3
   (b) Did you include complete proofs of all theoretical results? [Yes] Please check our appendix
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Please check section 5 and E
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] Please check section 5 and E
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]