SIAM J. MATH. DATA SCI.
Vol. 3, No. 2, pp. 544–572

Binary Component Decomposition Part I: The Positive-Semidefinite Case*

Richard Kueng† and Joel A. Tropp†

Abstract. This paper studies the problem of decomposing a low-rank positive-semidefinite matrix into symmetric factors with binary entries, either $\{\pm 1\}$ or $\{0, 1\}$. This research answers fundamental questions about the existence and uniqueness of these decompositions. It also leads to tractable factorization algorithms that succeed under a mild deterministic condition. A companion paper addresses the related problem of decomposing a low-rank rectangular matrix into a binary factor and an unconstrained factor.

Key words. matrix factorization, cut polytope, elliptope, semidefinite programming

AMS subject classifications. Primary, 52A20, 15B48; Secondary, 15A21, 52B12, 90C27

DOI. 10.1137/19M1278612

1. Motivation and background.

Matrix factorization stands among the most fundamental methods for unsupervised data analysis. One of the main purposes of factorization is to identify latent structure in a matrix. Other applications include data compression, summarization, and visualization. In many situations, we need to place constraints on the factors appearing in the matrix decomposition. This step allows us to enforce prior knowledge about the process that generates the data, thereby enhancing our ability to detect structure.

Prominent examples of constrained matrix factorizations include independent component analysis [Com94], nonnegative matrix factorization [PT94], dictionary learning or sparse coding [OF96], and sparse principal component analysis [ZHT06]. These techniques arose in signal processing, environmental engineering, neuroscience, and statistics. This catalog hints at the wide compass of these ideas. It is a natural challenge to develop a rigorous theory that justifies and improves existing factorization models. Another valuable direction is to create new types of constrained factorizations. These problems not only have a deep intellectual appeal, but progress may eventually lead to new modes of data analysis.

The purpose of this paper and its companion [KT19] is to develop the theoretical foundations for *binary component decompositions*. That is, we are interested in matrix decompositions where one of the factors is required to take values in the set $\{\pm 1\}$ or in the set $\{0, 1\}$. These models are appropriate for applications where the latent factor reflects an exclusive choice. For instance, "on" and "off" in electrical engineering; "connected" or "disconnected"

*Received by the editors August 1, 2019; accepted for publication (in revised form) January 20, 2021; published electronically May 3, 2021.
https://doi.org/10.1137/19M1278612
Funding: This work was partially funded by ONR awards N00014-11-1002, N00014-17-12146, and N00014-18-12363. Additional support was provided by the Gordon & Betty Moore Foundation.

†Department of Computing + Mathematical Sciences, California Institute of Technology, Pasadena, CA 91125-5000 USA (rkueng@caltech.edu, jtropp@cms.caltech.edu).

in graph theory; "yes" and "no" in survey data; "like" and "dislike" in collaborative filtering; or "active" and "inactive" in genomics.

In this first paper, we study factorization of a low-rank correlation matrix into symmetric binary factors. We focus on core questions about the existence and uniqueness, and we develop efficient algorithms for computing these factorizations in the noiseless setting. We also describe a stylized application in massive MIMO communications. In the companion paper [KT19], we build on these ideas to develop an asymmetric factorization of a low-rank data matrix into a binary factor and an unconstrained factor, and we present some robustness results.

1.1. Notation. We use standard notation. Scalars are indicated by lowercase Roman or Greek letters (x, ξ); lowercase bold letters $(\boldsymbol{x}, \boldsymbol{\xi})$ are (column) vectors; and uppercase bold letters $(\boldsymbol{X}, \boldsymbol{\Xi})$ are matrices. Calligraphic letters $(\mathcal{X})$ are reserved for sets.

Throughout, n is a fixed natural number. We work in the real linear space $\mathbb{R}^n$ equipped with the standard inner product and the associated norm topology. The symbol $^{\mathsf{t}}$ denotes the transpose of a vector or matrix. The standard basis in $\mathbb{R}^n$ is the set $\{\mathbf{e}_1, \dots, \mathbf{e}_n\}$. We write $\mathbf{e}$ for the vector of ones; its dimension is determined by context. The symbol $\odot$ denotes the Schur (i.e., componentwise) product of vectors. The closed and open probability simplices are

$$\Delta_r = \left\{ \boldsymbol{\tau} \in \mathbb{R}^r : \tau_i \geq 0, \sum\nolimits_{i=1}^r \tau_i = 1 \right\} \quad \text{and} \quad \Delta_r^+ = \left\{ \boldsymbol{\tau} \in \mathbb{R}^r : \tau_i > 0, \sum\nolimits_{i=1}^r \tau_i = 1 \right\}.$$

These sets parameterize the coefficients in a convex combination.

The real linear space $\mathbb{H}_n$ consists of symmetric $n \times n$ matrices with real entries. We write $\mathbf{I}$ for the identity matrix; its dimension is determined by context. A positive-semidefinite (psd) matrix is a symmetric matrix with nonnegative eigenvalues. The statement $\boldsymbol{X} \succcurlyeq \mathbf{0}$ means that $\boldsymbol{X}$ is psd. We often invoke the following fundamental property of psd matrices.

Fact 1.1 (conjugation rule). *Conjugation respects the semidefinite order.*

1. *If* $\boldsymbol{X} \succcurlyeq \mathbf{0}$, *then* $\boldsymbol{K}\boldsymbol{X}\boldsymbol{K}^{\mathsf{t}} \succcurlyeq \mathbf{0}$ *for each matrix* $\boldsymbol{K}$ *with compatible dimensions.*
2. *If* $\boldsymbol{K}$ *has full column rank and* $\boldsymbol{K}\boldsymbol{X}\boldsymbol{K}^{\mathsf{t}} \succcurlyeq \mathbf{0}$, *then* $\boldsymbol{X} \succcurlyeq \mathbf{0}$.

1.2. Background. Our point of departure is the famous eigenvalue decomposition. Let $\boldsymbol{A}$ be a rank-r *correlation matrix*. That is, $\boldsymbol{A}$ is a rank-r psd matrix with all diagonal entries equal to one. We can always write this matrix in the form

$$\boldsymbol{A} = \sum\nolimits_{i=1}^r \lambda_i \boldsymbol{u}_i \boldsymbol{u}_i^{\mathsf{t}}, \quad \text{where} \quad \operatorname{diag}(\boldsymbol{A}) = \mathbf{e}. \tag{1.1}$$

In this expression, $\{\boldsymbol{u}_1, \dots, \boldsymbol{u}_r\} \subset \mathbb{R}^n$ is an orthonormal family of eigenvectors associated with the positive eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_r > 0$. Eigenvalue decompositions are a basic tool in data analysis because of the connection with principal component analysis [Jol02].

In spite of the significance and elegance of the decomposition (1.1), it suffers from several debilities. First, we cannot impose extra conditions on the eigenvectors to enforce prior knowledge about the data. Second, the eigenvectors are a mathematical abstraction, so they often lack a meaning or interpretation. Moreover, in applications it is often hard to argue that the data was generated from orthogonal components.

These limitations are widely known and have spurred the development of matrix decompositions that evince other types of structure. For example, we may seek a decomposition of the form (1.1) where the vectors $\boldsymbol{u}_i$ are discrete, sparse, or nonnegative.

The most popular computational approach for structured factorization alternates between optimization and deflation steps. For example, see [OP83a, Kol98, FK99, Tem03, AN06, Wit10, Jag11, Bac13, Ude15, Bru17]. The idea is to iteratively find the structured components and extract them one by one. The optimization step (attempts) to identify an individual component by solving a problem like

$$\text{maximize} \quad \boldsymbol{x}^{\mathsf{t}} \boldsymbol{A} \boldsymbol{x} \quad \text{subject to} \quad \boldsymbol{x} \text{ is "structured."} \tag{1.2}$$

Given an approximate solution $\boldsymbol{x}$, one tries to remove its contribution to the target matrix. For instance, one might make the update

$$\boldsymbol{A} \mapsto \boldsymbol{A} - \lambda \boldsymbol{x}\boldsymbol{x}^{\mathsf{t}}, \quad \text{where} \quad \lambda = \boldsymbol{x}^{\mathsf{t}} \boldsymbol{A} \boldsymbol{x} / \left\| \boldsymbol{x} \right\|^2. \tag{1.3}$$

Other deflation techniques include matching pursuit algorithms [DT96] and conditional gradient methods [Jag11].

For several reasons, deflation-based factorization algorithms often lack strong guarantees. First, the optimization step (1.2) may not identify a structured component that actually appears in the matrix. Furthermore, it is usually computationally hard to solve (1.2) to optimality [AN06, GV18]. We refer to section 8 for a more detailed discussion, as well as an overview of other algorithms for discrete matrix factorization.

2. Sign component decomposition and binary component decomposition.

This section introduces the two matrix factorizations that we will study in this paper, the sign component decomposition (subsection 2.1) and the binary component decomposition (subsection 2.2). It also presents our main results about situations when we can compute these factorizations with polynomial-time algorithms. We conclude with an outline of this paper (subsection 2.3).

2.1. Sign component decomposition.

In this work, we study low-rank matrix factorization models where the underlying components are binary-valued and need not be orthogonal. We begin with the case where the entries of the components are restricted to the set $\{\pm 1\}$. In subsection 2.2, we discuss an alternative model where the entries are restricted to $\{0, 1\}$.

For a natural number r, we consider the problem of decomposing a rank-r correlation matrix $\boldsymbol{A} \in \mathbb{H}_n$ as a proper[1] convex combination of rank-one sign matrices:

$$\boldsymbol{A} = \sum\nolimits_{i=1}^{r} \tau_i \boldsymbol{s}_i \boldsymbol{s}_i^{\mathsf{t}}, \quad \text{where} \quad \boldsymbol{s}_i \in \{\pm 1\}^r \quad \text{and} \quad (\tau_1, \dots, \tau_r) \in \Delta_r^+. \tag{2.1}$$

Equivalently, we may write the decomposition (2.1) as a matrix factorization:

$$\boldsymbol{A} = \boldsymbol{S} \operatorname{diag}(\boldsymbol{\tau}) \boldsymbol{S}^{\mathsf{t}}, \quad \text{where} \quad \boldsymbol{S} = \begin{bmatrix} \boldsymbol{s}_1 & \dots & \boldsymbol{s}_r \end{bmatrix} \in \{\pm 1\}^{n \times r}, \quad \boldsymbol{\tau} = (\tau_1, \dots, \tau_r) \in \Delta_r^+. \tag{2.2}$$

Note that the right-hand side of (2.1) always yields a correlation matrix. See Figure 1 for a schematic. We refer to the factorization (2.1)–(2.2) as a *sign component decomposition* of the correlation matrix $\boldsymbol{A}$. The ± 1-valued vectors $\boldsymbol{s}_i$ are called *sign components*, and they may be correlated with each other. Altogether, these properties give the factorization a combinatorial flavor, rather than a geometric one.

[1] A *proper* convex combination has strictly positive coefficients.

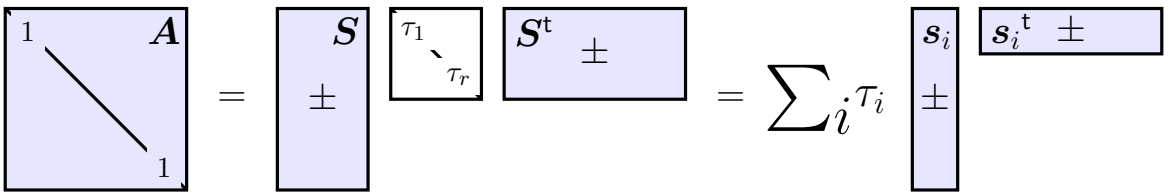

Figure 1. *Sign component decomposition. The sign component decomposition (2.1)–(2.2) expresses a correlation matrix $\boldsymbol{A}$ as a proper convex combination of rank-one sign matrices.*

2.1.1. Schur independence and geometry. Although the sign component decomposition may appear to be combinatorially intricate, we can compute it efficiently for a surprisingly large class of instances. This positive outcome stems from remarkable geometric properties of the set of correlation matrices. The following definition from [LP96] is central to our program.

Definition 2.1 (Schur independence of sign vectors). *A set $\{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\} \subseteq \{\pm 1\}^n$ of sign vectors is* Schur independent *if the linear hull of all pairwise Schur products has the maximal dimension:*

$$\dim \operatorname{span}\big\{\boldsymbol{s}_i \odot \boldsymbol{s}_j : 1 \le i, j \le r\big\} = \tbinom{r}{2} + 1.$$

Equivalently, the multiset $\{\mathbf{e}\} \cup \{\boldsymbol{s}_i \odot \boldsymbol{s}_j : 1 \le i < j \le r\} \subset \mathbb{R}^n$ must be linearly independent.

Here are some simple observations. If a set is Schur independent, so is every subset. Schur independence of a set is unaffected if we flip the sign of any subset of the vectors. Last, it is computationally easy to check if a set of sign vectors is Schur independent.

We can interpret Definition 2.1 as a "general position" property for sign vectors. A Schur independent multiset is always linearly independent (Lemma 4.7), but the converse is not true in general. Indeed, the cardinality r of a Schur-independent collection of sign vectors in $\mathbb{R}^n$ must satisfy the bound

$$r \le \tfrac{1}{2}\big(1 + \sqrt{8n - 7}\big); \tag{2.3}$$

see [LP96, eq. (3.6)]. When r meets the threshold (2.3), most collections of r sign vectors are Schur independent. Indeed, a randomly chosen family of sign vectors is Schur independent with overwhelming probability. Here is a basic result in this direction [Tro18, Thm. 2.9].

Fact 2.2 (Tropp). *Suppose that the vectors $\boldsymbol{s}_1, \dots, \boldsymbol{s}_r$ are drawn independently and uniformly at random from $\{\pm 1\}^n$. Then $\{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\}$ is Schur independent with probability at least $1 - r^2 \exp(-n/r^2)$.*

See [Tro18, Thms. 2.10, 2.11] for other probability models and significant improvements.

Although Schur independence may seem alien at first sight, it appears naturally when we adopt a geometric perspective on the sign component decomposition. The correlation matrix $\boldsymbol{A}$ is contained in the convex hull $\mathcal{F}$ of its sign components. This convex hull is, in turn, a proper convex subset of the (convex) set of all correlation matrices $\mathcal{E}_n$:

$$\boldsymbol{A} = \sum\nolimits_{i=1}^{r} \tau_i \boldsymbol{s}_i \boldsymbol{s}_i^{\mathsf{t}} \subset \underbrace{\operatorname{conv}\big\{\boldsymbol{s}_1\boldsymbol{s}_1^{\mathsf{t}}, \dots, \boldsymbol{s}_r\boldsymbol{s}_r^{\mathsf{t}}\big\}}_{\mathcal{F}} \subset \underbrace{\{\boldsymbol{X} \in \mathbb{H}_n : \operatorname{diag}(\boldsymbol{X}) = \mathbf{e} \text{ and } \boldsymbol{X} \succcurlyeq \mathbf{0}\}}_{\mathcal{E}_n}.$$

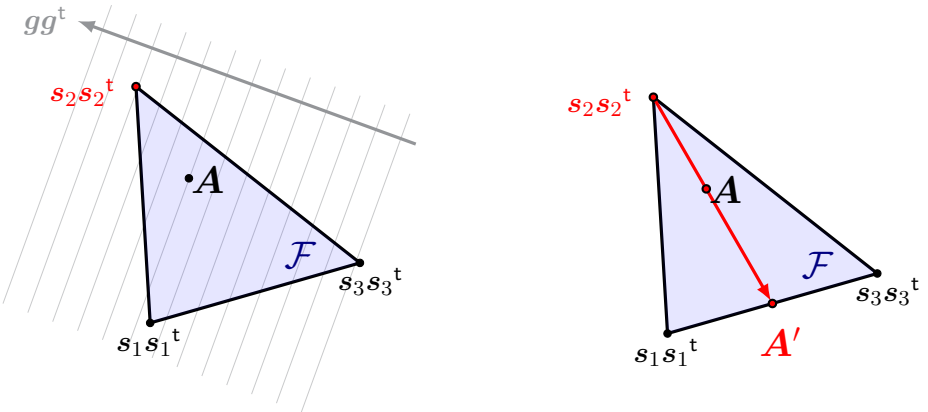

Figure 2. *Illustration of Algorithm 2.1. The matrix $\boldsymbol{A}$ belongs to the relative interior of the simplex $\mathcal{F} = \operatorname{conv}\{\boldsymbol{s}_1\boldsymbol{s}_1^{\mathsf{t}}, \boldsymbol{s}_2\boldsymbol{s}_2^{\mathsf{t}}, \boldsymbol{s}_3\boldsymbol{s}_3^{\mathsf{t}}\}$.* Left*: Random optimization over the simplex $\mathcal{F}$ identifies an extreme point with probability one. In this diagram, maximizing the linear functional $\boldsymbol{X} \mapsto \operatorname{trace}(\boldsymbol{g}\boldsymbol{g}^{\mathsf{t}}\boldsymbol{X})$ over $\mathcal{F}$ locates the rank-one matrix $\boldsymbol{s}_2\boldsymbol{s}_2^{\mathsf{t}}$.* Right*: To remove the contribution of the rank-one matrix $\boldsymbol{s}_2\boldsymbol{s}_2^{\mathsf{t}}$ from the matrix $\boldsymbol{A}$, we traverse the ray from the rank-one matrix through the matrix $\boldsymbol{A}$ until we arrive at a facet of $\mathcal{F}$. The terminus $\boldsymbol{A}'$ of the ray is a proper convex combination of the remaining rank-one matrices.*

Schur independence of the sign components $\{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\}$ implies that the set $\mathcal{F}$ is a *simplicial face* of $\mathcal{E}_n$. We refer to subsection 3.5 for a detailed statement and context.

2.1.2. Computing a sign component decomposition. The main outcome of this paper is an efficient algorithm for computing the sign component decomposition of a rich class of correlation matrices. Schur independence of the sign components is necessary and sufficient for this method to operate. Geometrically, this property ensures that the matrix $\boldsymbol{A} = \sum_{i=1}^{r} \tau_i \boldsymbol{s}_i\boldsymbol{s}_i^{\mathsf{t}}$ is contained in a simplicial face $\mathcal{F} = \{\boldsymbol{s}_1\boldsymbol{s}_1^{\mathsf{t}}, \dots, \boldsymbol{s}_r\boldsymbol{s}_r^{\mathsf{t}}\}$ of the set $\mathcal{E}_n$ of correlation matrices. Since $\mathcal{F}$ lies on the boundary of $\mathcal{E}_n$, it is possible to isolate this simplicial face with a linear separator. Indeed, if $\boldsymbol{P} \in \mathbb{H}_n$ is the orthogonal projector onto the range of $\boldsymbol{A}$, then

$$\mathcal{F} = \{\boldsymbol{X} \in \mathbb{H}_n : \operatorname{diag}(\boldsymbol{X}) = \mathbf{e},\ \operatorname{tr}(\boldsymbol{P}\boldsymbol{X}) = n \text{ and, } \boldsymbol{X} \succcurlyeq \mathbf{0}\}.$$

This claim is established in Proposition 3.6 and allows us to efficiently optimize linear functionals over $\mathcal{F}$ via semidefinite programming.

Optimizing a random functional $\boldsymbol{g}^{\mathsf{t}}\boldsymbol{X}\boldsymbol{g}$ almost surely locates an extreme point $\boldsymbol{X}_\star = \boldsymbol{s}_k\boldsymbol{s}_k^{\mathsf{t}}$ of the face $\mathcal{F}$. In this case, by factorizing $\boldsymbol{X}_\star$, we can identify a sign component of the matrix $\boldsymbol{A}$. We may view this approach as a provably correct variant of the optimization step (1.2) in a deflation procedure. Subsequently, we can remove the contribution of $\boldsymbol{s}_k\boldsymbol{s}_k^{\mathsf{t}}$ to $\boldsymbol{A}$ in a fashion that respects the geometric structure and iterate. This procedure is summarized in Algorithm 2.1 and we refer to Figure 2 for a visual illustration.

Theorem I (sign component decomposition). *Let $\boldsymbol{A} \in \mathbb{H}_n$ be a rank-r correlation matrix that admits a sign component decomposition (2.1)–(2.2) where the set $\{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\}$ of sign components is Schur independent. Then the sign component decomposition is uniquely determined up to trivial symmetries, and Algorithm 2.1 computes the decomposition in time polynomial in n.*

Algorithm 2.1. *Sign component decomposition (2.1) of a matrix with Schur independent components.*

Implements the procedure from subsection 3.7.

Input: Rank-r correlation matrix $\boldsymbol{A} \in \mathbb{H}_n$ that satisfies (2.1) with Schur independent sign components
Output: Sign components $\{\tilde{\boldsymbol{s}}_1, \dots, \tilde{\boldsymbol{s}}_r\} \subseteq \{\pm 1\}^n$ and convex coefficients $\tilde{\boldsymbol{\tau}} \in \Delta_r^+$, where $\boldsymbol{A} = \sum_{i=1}^r \tilde{\tau}_i \tilde{\boldsymbol{s}}_i \tilde{\boldsymbol{s}}_i^{\mathsf{t}}$

function SignComponentDecomposition($\boldsymbol{A}$)
  $[n, \sim] \leftarrow \texttt{size}(\boldsymbol{A})$ and $r \leftarrow \operatorname{rank}(\boldsymbol{A})$
  for $i = 1$ to $(r-1)$ **do**
    $\boldsymbol{U} \leftarrow \texttt{orth}(\boldsymbol{A})$ ▷ Find a basis for the range of $\boldsymbol{A}$
    $\boldsymbol{g} \leftarrow \texttt{randn}(n, 1)$ ▷ Draw a random direction
    Find the solution $\boldsymbol{X}_\star$ to the semidefinite program ▷ Step 1

$$\underset{\boldsymbol{X} \in \mathbb{H}_n}{\text{maximize}} \quad \boldsymbol{g}^{\mathsf{t}} \boldsymbol{X} \boldsymbol{g} \quad \text{subject to} \quad \operatorname{trace}\left(\boldsymbol{U}^{\mathsf{t}} \boldsymbol{X} \boldsymbol{U}\right) = n, \operatorname{diag}(\boldsymbol{X}) = \mathbf{e}, \text{ and } \boldsymbol{X} \succcurlyeq \mathbf{0}$$

    Factorize the rank-one matrix $\boldsymbol{X}_\star = \tilde{\boldsymbol{s}}_i \tilde{\boldsymbol{s}}_i^{\mathsf{t}}$ ▷ Extract a sign component
    Find the solution $\zeta_\star$ to the semidefinite program ▷ Step 2

$$\underset{\zeta \in \mathbb{R}}{\text{maximize}} \quad \zeta \quad \text{subject to} \quad \zeta \boldsymbol{A} + (1 - \zeta) \boldsymbol{X}_\star \succcurlyeq \mathbf{0}$$

    $\boldsymbol{A} \leftarrow \zeta_\star \boldsymbol{A} + (1 - \zeta_\star) \boldsymbol{X}_\star$ ▷ Step 3
  Factorize the rank-one matrix $\boldsymbol{A} = \tilde{\boldsymbol{s}}_r \tilde{\boldsymbol{s}}_r^{\mathsf{t}}$ ▷ $\operatorname{rank}(\boldsymbol{A}) = 1$ in final iteration
  Find the solution $\tilde{\boldsymbol{\tau}} \in \Delta_r^+$ to the linear system ▷ Step 4

$$\boldsymbol{A} = \sum\nolimits_{i=1}^r \tilde{\tau}_i \tilde{\boldsymbol{s}}_i \tilde{\boldsymbol{s}}_i^{\mathsf{t}}$$

The uniqueness claim in Theorem I is established in Theorem 3.4. The computational claim is the content of Theorem 3.7.

Theorem I only asserts that we can factorize a low-rank matrix. We report some ideas for overcoming this difficulty in the companion paper [KT19], but the fundamental problem of approximate sign component decomposition of a noisy matrix remains open. Indeed, Theorem I relies on geometric properties that are not stable under perturbation of the input matrix. We hope to address this serious practical issue in future work.

2.1.3. Remarks on implementation. As stated, Algorithm 2.1 involves $2(r-1)$ semidefinite programs over the $n \times n$ matrices. Using randomized dimension reduction, we can develop an equivalent procedure that optimizes over $r \times r$ matrices instead. Algorithm SM1.1, detailed in the supplement, reduces the arithmetic cost to $\mathcal{O}(n^3 \operatorname{polylog}(r))$, which is comparable to a single dense eigenvalue decomposition.

We can implement Algorithm 2.1 or Algorithm SM1.1 with a general-purpose SDP solver (based on an interior-point method, say). At present, this approach can factorize a matrix with dimension n in the low 1000s. Scalable semidefinite programming algorithms, such as the Burer–Monteiro method [BM03, BBV16] or SketchyCGAL [YTF+19], can handle somewhat larger problems, but storage of the dense correlation matrix $\boldsymbol{A}$ remains an issue.

Remark 2.3 (certificates for correctness and uniqueness). Algorithm 2.1 can detect its own failure in an online fashion. It reveals sign components one by one. This allows for repeatedly checking Schur independence and failing loudly if Schur independence is violated. Schur independence of all sign components also implies that the obtained factorization is unique up to trivial ambiguities.

2.2. Binary component decomposition. Sign component decomposition provides a foundation for computing other types of binary factorizations. In particular, we can also study models where the components take values in the set $\{0, 1\}$. Let $\boldsymbol{H} \in \mathbb{H}_n$ be a psd matrix. Our goal is to find a representation

$$\boldsymbol{H} = \sum\nolimits_{i=1}^{r} \tau_i \boldsymbol{z}_i \boldsymbol{z}_i^{\mathsf{t}} \quad \text{where,} \quad \boldsymbol{z}_i \in \{0,1\}^n \quad \text{and} \quad (\tau_1, \dots, \tau_r) \in \Delta_r^+. \tag{2.4}$$

Equivalently, we may write the decomposition (2.4) as a matrix factorization:

$$\boldsymbol{A} = \boldsymbol{Z}\, \operatorname{diag}(\boldsymbol{\tau})\, \boldsymbol{Z}^{\mathsf{t}}, \quad \text{where} \quad \boldsymbol{Z} = \begin{bmatrix} \boldsymbol{z}_1 & \dots & \boldsymbol{z}_r \end{bmatrix} \in \{0,1\}^{n \times r}, \quad \boldsymbol{\tau} = (\tau_1, \dots, \tau_r) \in \Delta_r^+. \tag{2.5}$$

We refer to (2.4)–(2.5) as a *binary component decomposition* of the matrix $\boldsymbol{H}$. The vectors $\boldsymbol{z}_i$ are called *binary components*.

We can connect the binary component decomposition with the sign component decomposition by a simple device. Just observe that there is an affine map that places binary vectors and sign vectors in one-to-one correspondence:

$$\boldsymbol{F} : \{0,1\}^n \to \{\pm 1\}^n, \quad \text{where} \quad \boldsymbol{F} : \boldsymbol{z} \mapsto 2\boldsymbol{z} - \mathbf{e} \quad \text{and} \quad \boldsymbol{F}^{-1} : \boldsymbol{s} \mapsto \tfrac{1}{2}(\boldsymbol{s} + \mathbf{e}). \tag{2.6}$$

Owing to the correspondence (2.6), Schur independence of sign vectors begets a concept of Schur independence for binary vectors.

Definition 2.4 (Schur independence of binary vectors). *Let $\boldsymbol{z}_0 = \mathbf{e}$. A set $\{\boldsymbol{z}_1, \dots, \boldsymbol{z}_r\} \subseteq \{0,1\}^n$ of binary vectors is* Schur independent *if*

$$\dim \operatorname{span}\big\{ \boldsymbol{z}_i \odot \boldsymbol{z}_j : 0 \le i, j \le r \big\} = \tbinom{r}{2} + 1.$$

Proposition 6.3 describes the precise relationship between the two notions of Schur independence. The correspondence (2.6) also allows us to reduce binary component decomposition to sign component decomposition. The following result is a (nontrivial) corollary of Theorem I.

Theorem II (binary component decomposition). *Let $\boldsymbol{H} \in \mathbb{H}_n$ be a rank-r psd matrix that admits a binary component decomposition (2.4)–(2.5) where the set $\{\boldsymbol{z}_1, \dots, \boldsymbol{z}_r\}$ of binary components is Schur independent. Then the binary component decomposition is uniquely determined up to trivial symmetries, and Algorithm 2.2 computes the decomposition in time polynomial in n.*

See subsection 6.4 for the proof.

The binary component decomposition (2.4) is closely related to the (symmetric) cut decomposition [FK99, AN06]. In general, cut decompositions appear to involve challenging combinatorial optimization problems. Viewed from this angle, it seems surprising that binary component decompositions are unique and efficiently computable. See section 8 for a discussion.

Algorithm 2.2. *Binary component decomposition (2.4) of a matrix with Schur independent components.*

Implements the procedure from subsection 6.4.

Input: Rank-r symmetric matrix $\boldsymbol{H} \in \mathbb{H}_n$ that satisfies (2.4) with Schur independent binary components.

Output: Binary components $\{\tilde{\boldsymbol{z}}_1, \dots, \tilde{\boldsymbol{z}}_r\} \subseteq \{0,1\}^n$ and convex coefficients $\tilde{\boldsymbol{\tau}} \in \Delta_r$, where $\boldsymbol{H} = \sum_{i=1}^r \tilde{\tau}_i \, \tilde{\boldsymbol{z}}_i \tilde{\boldsymbol{z}}_i^{\mathsf{t}}$

function BINARYCOMPONENTDECOMPOSITION($\boldsymbol{H}$)

Find the solution $\boldsymbol{A} \in \mathbb{H}_n$ to the linear system

$$\operatorname{diag}(\boldsymbol{X}) = \mathbf{e} \quad \text{and} \quad \boldsymbol{R}(4\boldsymbol{H} - \boldsymbol{X})\boldsymbol{R} = \mathbf{0}, \quad \text{where} \quad \boldsymbol{R} = \mathbf{I} - n^{-1}\mathbf{e}\mathbf{e}^{\mathsf{t}}$$

Apply Algorithm Algorithm 2.1 to $\boldsymbol{A}$ to obtain sign components $\tilde{\boldsymbol{s}}_1, \dots, \tilde{\boldsymbol{s}}_r$ and convex coefficients $\tilde{\boldsymbol{\tau}} \in \Delta_r^+$

Find the solution $\boldsymbol{\xi} \in \mathbb{R}^r$ to the linear system ▷ Resolve sign ambiguities

$$n \sum\nolimits_{i=1}^r \tilde{\tau}_i \xi_i \tilde{\boldsymbol{s}}_i = (4\boldsymbol{H} - \boldsymbol{A})\mathbf{e} - 2n \operatorname{trace}(\boldsymbol{H})\mathbf{e}$$

Set $\tilde{\boldsymbol{z}}_i = \frac{1}{2}(\xi_i \tilde{\boldsymbol{s}}_i + \mathbf{e})$ for each index i

It is worth mentioning that the regularity condition for binary components differs slightly from its counterpart for sign components: The vector $\mathbf{e}$ features in Definition 2.4 but not in Definition 2.1. This modification imposes slightly more stringent conditions on binary components. It arises from the fact that the two decompositions enjoy different symmetries: sign vectors are invariant under flipping the global sign, while binary vectors are not.

2.3. Roadmap. Section 3 discusses the problems of existence, uniqueness, and computability of sign component decompositions at a high level. Section 4 elaborates on the geometry of the set of correlation matrices and its implications for sign component decomposition. Section 5 proves that Algorithm 2.1 computes a sign component decomposition. Section 6 treats the binary component decomposition. Afterward, in section 7, we present a stylized application to massive MIMO communication. Section 8 details related work.

3. Existence, uniqueness, and computation.

This section expands on the geometric formulation of the sign component decomposition. This perspective leads to our main results on existence, uniqueness, and computability.

3.1. Questions. We focus on three fundamental problems raised by the definition (2.1)–(2.2) of the sign component decomposition.

1. **Existence.** Which correlation matrices admit a sign component decomposition?
2. **Uniqueness.** When is the sign component decomposition unique up to symmetries?
3. **Computation.** How can we find a sign component decomposition in polynomial time?

The rest of this section summarizes our answers to these questions. To make the narrative more kinetic, we postpone some standard definitions and the details of the analysis to subsequent sections. While the first two problems reduce to basic geometric considerations, our

investigation of the third question pilots us into more interesting territory.

There is also a fourth fundamental problem.

4. **Robustness.** Can we find a sign component decomposition from a noisy observation? We do not treat this question here, but we present some limited results for a closely related decomposition in the companion work [KT19]. Understanding robustness is a critical topic for future research.

3.2. Existence of the sign component decomposition.

The first order of business is to delineate circumstances in which a correlation matrix admits a sign component decomposition.

To that end, we introduce the *elliptope*, the set of all correlation matrices with fixed dimension:

$$\mathcal{E}_n = \{\boldsymbol{X} \in \mathbb{H}_n : \operatorname{diag}(\boldsymbol{X}) = \mathbf{e} \text{ and } \boldsymbol{X} \succcurlyeq \mathbf{0}\}.$$

The geometry of the elliptope plays a central role in our development, so we take note of some basic properties. The elliptope $\mathcal{E}_n$ is a compact convex subset of $\mathbb{H}_n$, and we can optimize a linear functional over the elliptope using a simple semidefinite program. Among other things, the elliptope $\mathcal{E}_n$ contains each rank-one sign matrix $\boldsymbol{s}\boldsymbol{s}^{\mathsf{t}}$ generated by a sign vector $\boldsymbol{s} \in \{\pm 1\}^n$. In fact, each rank-one sign matrix is an extreme point of the elliptope.

Next, let us construct the set of correlation matrices that admit a sign component decomposition. The (signed) *cut polytope* is the convex hull of the rank-one sign matrices:

$$\mathcal{C}_n = \operatorname{conv}\left\{\boldsymbol{s}\boldsymbol{s}^{\mathsf{t}} : \boldsymbol{s} \in \{\pm 1\}^n\right\} \subset \mathbb{H}_n. \tag{3.1}$$

It is easy to verify that the extreme points of the cut polytope are precisely the rank-one sign matrices. Since each rank-one sign matrix belongs to the elliptope, convexity ensures that the cut polytope is strictly contained in the elliptope: $\mathcal{C}_n \subset \mathcal{E}_n$. In view of these relationships, we can think about the elliptope as a semidefinite relaxation of the cut polytope.

The next statement is an immediate consequence of (2.1) and (3.1).

Proposition 3.1 (sign component decomposition: existence). *A correlation matrix $\boldsymbol{A} \in \mathcal{E}_n$ admits a sign component decomposition (2.1) if and only if $\boldsymbol{A} \in \mathcal{C}_n$.*

This simple result masks the true difficulty of the problem because the cut polytope is a very complicated object. In fact, it is computationally hard just to decide whether a given correlation matrix belongs to the cut polytope [DL97].

3.3. Symmetries of the sign component decomposition.

Proposition 3.1 tells us that each matrix in the cut polytope admits a sign component decomposition. The next challenge is to understand when this decomposition is determined uniquely.

First, observe that each sign component decomposition $\boldsymbol{A} = \sum_{i=1}^{r} \tau_i \boldsymbol{s}_i \boldsymbol{s}_i^{\mathsf{t}}$ has a parametric representation of the form $\{(\tau_i, \boldsymbol{s}_i) : i = 1, \dots, r\}$. In this representation, r is a natural number, $(\tau_1, \dots, \tau_r) \in \Delta_r^+$, and the sign components $\boldsymbol{s}_i \in \{\pm 1\}^n$. But there is no way to distinguish an ordering of the pairs ($i \mapsto \pi(i)$ for a permutation π) or to distinguish the global sign of a sign component ($\boldsymbol{s}_i \mapsto \xi_i \boldsymbol{s}_i$ for $\xi_i \in \{\pm 1\}$). Therefore, we regard two parametric representations as *equivalent* if they have the same number of terms and the terms coincide up to permutations and sign flips.

In summary, a correlation matrix has a *unique* sign component decomposition if the parametric representation of every possible sign component decomposition belongs to the same equivalence class.

3.4. Uniqueness of the sign component decomposition. Geometrically, the sign component decomposition (2.1) is a representation of a matrix $\boldsymbol{A} \in \mathcal{C}_n$ as a proper convex combination of the extreme points of the cut polytope, namely the rank-one sign matrices. The representation is unique if and only if the participating extreme points generate a face of the cut polytope that is also a simplex (i.e., the convex hull of an affinely independent point set).

Proposition 3.2 (sign component decomposition: uniqueness). *A matrix $\boldsymbol{A} \in \mathcal{C}_n$ admits a unique sign component decomposition (2.1) if and only if $\boldsymbol{A}$ belongs to the relative interior of a simplicial face of the cut polytope $\mathcal{C}_n$.*

See subsection 4.2 for the definition of a simplicial face; Proposition 3.2 follows from the discussion there.

Unfortunately, there is no simple or computationally tractable description of the simplicial faces of the cut polytope [DL97]. As a consequence, we cannot expect to produce a sign component decomposition of a general element of the cut polytope, even when the decomposition is uniquely determined.

Instead, let us focus on simplicial faces of the elliptope that are generated by rank-one sign matrices. These distinguished faces are always simplicial faces of the cut polytope because $\mathcal{C}_n \subset \mathcal{E}_n$ and the rank-one sign matrices are extreme points of both sets. Thus, Proposition 3.2 has the following consequence.

Corollary 3.3 (sign component decomposition: sufficient condition for uniqueness). *For a set $\mathcal{S} = \{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\} \subseteq \{\pm 1\}^n$ of sign vectors, suppose that $\mathcal{F} = \operatorname{conv}\{\boldsymbol{s}\boldsymbol{s}^{\mathsf{t}} : \boldsymbol{s} \in \mathcal{S}\}$ is a simplicial face of the elliptope $\mathcal{E}_n$. If $\boldsymbol{A}$ belongs to the relative interior of $\mathcal{F}$, then $\boldsymbol{A}$ admits a unique sign component decomposition (2.1).*

See subsection 4.4 for further details.

3.5. Simplicial faces of the elliptope. This is where things get interesting. Corollary 3.3 suggests that we shift our attention to those correlation matrices that belong to a simplicial face of the elliptope that is generated by rank-one sign matrices. This class of matrices admits a beautiful characterization.

Theorem 3.4 (simplicial faces of the elliptope: characterization). *Let $\mathcal{S} = \{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\} \subseteq \{\pm 1\}^n$ be a set of sign vectors. The following are equivalent:*

(1) *The set $\mathcal{S}$ of sign vectors is Schur independent.*

(2) *The set $\mathcal{F} = \operatorname{conv}\{\boldsymbol{s}\boldsymbol{s}^{\mathsf{t}} : \boldsymbol{s} \in \mathcal{S}\}$ is a simplicial face of the elliptope $\mathcal{E}_n$.*

Either condition implies that each correlation matrix in the relative interior of $\mathcal{F}$ has a unique sign component decomposition (2.1).

The implication (1) $\Rightarrow$ (2) was established by Laurent and Poljak [LP96]. The reverse direction (2) $\Rightarrow$ (1) is new; see subsection 4.4 for the proof. The last statement is the content of Corollary 3.3.

It is instructive to combine this insight with Fact 2.2. As soon as r satisfies $r^2 \log(r^2) < n$, almost all families of r sign vectors in $\mathbb{R}^n$ are Schur independent. The convex hull of the

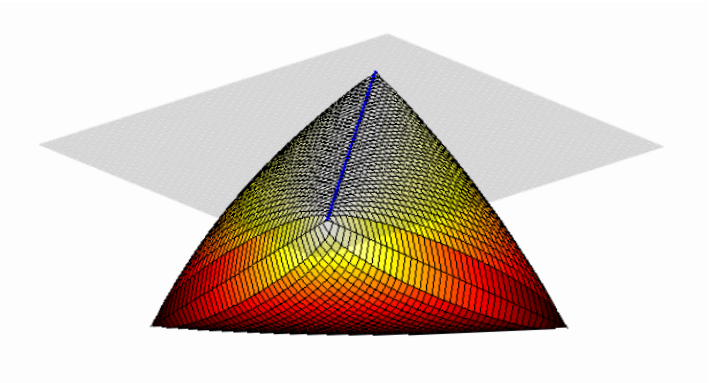

Figure 3. *Exposing a simplicial face of the elliptope. The hyperplane (gray) separates a one-dimensional simplicial face (blue) from the elliptope $\mathcal{E}_3$ (orange).*

associated rank-one sign matrices forms a simplicial face of the elliptope. Every correlation matrix in the relative interior of this face admits a unique sign component decomposition. The problem is how to find the decomposition.

Remark 3.5 (other kinds of simplicial faces). The elliptope has simplicial faces that are not described by Theorem 3.4. Indeed, for $n \geq 5$, the elliptope $\mathcal{E}_n$ has edges that are not generated as the convex hull of two rank-one sign matrices; see [LP96, Example 3.3].

3.6. Separating faces from the elliptope. As we have seen, the elliptope has an enormous number of simplicial faces that are generated by rank-one sign matrices. Remarkably, we can produce an explicit linear functional that exposes any type of face. This construction allows us to optimize over (simplicial) faces, which is the core ingredient in our algorithm for sign component decomposition.

Proposition 3.6 (simplicial faces of the elliptope: finding a separator). *Let $\mathcal{F}$ be a face of the elliptope $\mathcal{E}_n$. Fix a point $\boldsymbol{A}$ in the relative interior of $\mathcal{F}$, and let $\boldsymbol{P} \in \mathbb{H}_n$ be the orthogonal projector onto the range of $\boldsymbol{A}$. Construct the linear functional*

$$\psi(\boldsymbol{X}) = n^{-1} \operatorname{trace}(\boldsymbol{P}\boldsymbol{X}) \quad \text{for } \boldsymbol{X} \in \mathbb{H}_n.$$

Then ψ exposes the face $\mathcal{F}$ of the elliptope $\mathcal{E}_n$. That is,

$$\psi(\boldsymbol{X}) \leq 1 \text{ for all } \boldsymbol{X} \in \mathcal{E}_n \quad \text{and} \quad \mathcal{F} = \{\boldsymbol{X} \in \mathcal{E}_n : \psi(\boldsymbol{X}) = 1\}.$$

See subsection 4.5 for the proof. See Figure 3 for an illustration.

3.7. Computing the sign component decomposition. With this preparation, we can fully understand Algorithm 2.1, which computes the sign component decomposition (2.1) of a correlation matrix whose sign components are Schur independent. The procedure is iterative, and can be regarded as an algorithmic implementation of Carathéodory's theorem [Sch14, Thm. 1.1.4] or a variant of the Grötschel–Lovász–Schrijver method [GLS93, Thm. 6.5.11].

Assume that we are given a correlation matrix $\boldsymbol{A} \in \mathcal{E}_n$ that admits a sign component decomposition with Schur independent sign components:

$$\boldsymbol{A} = \sum\nolimits_{i=1}^{r} \tau_i \boldsymbol{s}_i \boldsymbol{s}_i^{\mathsf{t}}, \quad \text{where} \quad \mathcal{S} = \{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\} \subseteq \{\pm 1\}^n \text{ is Schur independent.} \tag{3.2}$$

As usual, the coefficients $(\tau_1, \dots, \tau_r) \in \Delta_r^+$. The matrix $\boldsymbol{A}$ belongs to the relative interior of the set $\mathcal{F} = \operatorname{conv}\{\boldsymbol{s}\boldsymbol{s}^{\mathsf{t}} : \boldsymbol{s} \in \mathcal{S}\}$. Theorem 3.4 implies that $\mathcal{F}$ is a simplicial face of the elliptope $\mathcal{E}_n$ and the sign component decomposition of $\boldsymbol{A}$ is unique. Proposition 3.6 allows us to formulate optimization problems over the set $\mathcal{F}$.

The following procedure exploits these insights to identify the sign component decomposition of $\boldsymbol{A}$. Figure 2 illustrates the geometry, while Algorithm 2.1 provides pseudocode.

- **Step 0: Initialization.** Let $\boldsymbol{A} \in \mathcal{E}_n$ be a rank-r correlation matrix of form (3.2). Find the orthogonal projector $\boldsymbol{P} \in \mathbb{H}_n$ onto $\operatorname{range}(\boldsymbol{A})$, and set $\psi(\boldsymbol{X}) = n^{-1} \operatorname{trace}(\boldsymbol{P}\boldsymbol{X})$.
- **Step 1: Random optimization.** Draw a (standard normal) random vector $\boldsymbol{g} \in \mathbb{R}^n$. Solve the semidefinite program

$$\operatorname*{maximize}_{\boldsymbol{X} \in \mathbb{H}_n} \quad \boldsymbol{g}^{\mathsf{t}} \boldsymbol{X} \boldsymbol{g} \quad \text{subject to} \quad \psi(\boldsymbol{X}) = 1 \quad \text{and} \quad \boldsymbol{X} \in \mathcal{E}_n. \tag{3.3}$$

  According to Proposition 3.6, the constraint set is precisely the simplex $\mathcal{F}$. With probability one, the unique solution $\boldsymbol{X}_\star$ is an extreme point of $\mathcal{F}$. That is, $\boldsymbol{X}_\star = \boldsymbol{s}_k \boldsymbol{s}_k^{\mathsf{t}}$ for some index $1 \le k \le r$. By factorizing $\boldsymbol{X}_\star$, we can extract one sign component of the matrix $\boldsymbol{A}$.
- **Step 2: Deflation.** Draw a ray from the matrix $\boldsymbol{X}_\star$ through the matrix $\boldsymbol{A}$. Traverse the ray until we reach a facet $\mathcal{F}'$ of the simplex $\mathcal{F}$ by finding the solution $\zeta_\star$ of

$$\operatorname*{maximize}_{\zeta \in \mathbb{R}} \quad \zeta \quad \text{subject to} \quad \zeta \boldsymbol{A} + (1 - \zeta) \boldsymbol{X}_\star \in \mathcal{F}. \tag{3.4}$$

  In our context, this optimization problem can be simplified, as stated in Algorithm 2.1.
- **Step 3: Iteration.** Let $\boldsymbol{A}' = \zeta_\star \boldsymbol{A} + (1 - \zeta_\star) \boldsymbol{X}_\star$ be the terminus of the ray described in the last step. This construction ensures that $\boldsymbol{A}'$ belongs to the relative interior of the convex hull of all the rank-one sign matrices other than $\boldsymbol{X}_\star$. That is,

$$\boldsymbol{A}' \in \operatorname{relint} \operatorname{conv} \left\{ \boldsymbol{s}_i \boldsymbol{s}_i^{\mathsf{t}} : 1 \le i \le r \text{ and } i \ne k \right\} = \mathcal{F}'.$$

  Therefore, $\boldsymbol{A}'$ admits a sign component decomposition with Schur independent sign components. We may return to Step 0 and repeat the process with the rank-$(r-1)$ correlation matrix $\boldsymbol{A}'$. The total number of iterations is r.
- **Step 4: Coefficients.** Given the r computed sign components $\tilde{\boldsymbol{s}}_1, \dots, \tilde{\boldsymbol{s}}_r$, we identify the convex coefficients $\tilde{\boldsymbol{\tau}} \in \Delta_r^+$ by finding the unique solution to the linear system

$$\boldsymbol{A} = \sum\nolimits_{i=1}^{r} \tilde{\tau}_i \tilde{\boldsymbol{s}}_i \tilde{\boldsymbol{s}}_i^{\mathsf{t}}.$$

The following theorem states that this procedure yields a parametric representation of the unique sign component decomposition of the matrix $\boldsymbol{A}$.

Theorem 3.7 (analysis of Algorithm 2.1). *Let $\boldsymbol{A} \in \mathcal{E}_n$ be a correlation matrix that admits a sign component decomposition*

$$\boldsymbol{A} = \sum\nolimits_{i=1}^{r} \tau_i \boldsymbol{s}_i \boldsymbol{s}_i^{\mathsf{t}}, \quad \textit{where} \quad \boldsymbol{s}_i \in \{\pm 1\}^n \quad \textit{and} \quad (\tau_1, \dots, \tau_r) \in \Delta_r^+. \tag{3.5}$$

Assume that the set $\mathcal{S} = \{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\}$ of sign components is Schur independent. Then, with probability one, Algorithm 2.1 identifies the sign component decomposition of $\boldsymbol{A}$ up to trivial symmetries. That is, the output is an unordered set of pairs $\{(\tau_i, \xi_i \boldsymbol{s}_i) : 1 \le i \le r\}$, where $\xi_i \in \{\pm 1\}$ are signs.

Section 5 contains a full proof of this result.

4. Geometric aspects of the sign component decomposition. This section justifies the geometric claims propounded in the last section. The books [Roc70, HUL01, Bar02, Gru07, Sch14] serve as good references for convex geometry.

4.1. Faces of convex sets. In this section, we work in a finite-dimensional real vector space V, equipped with a norm topology. Let us begin with some basic facts about the boundary structure of a convex set.

Definition 4.1 (face). *Let $\mathcal{K}$ be a closed convex set in V. A* face *$\mathcal{F}$ of $\mathcal{K}$ is a convex subset of $\mathcal{K}$ for which*

$$\boldsymbol{x}, \boldsymbol{y} \in \mathcal{K} \quad \text{and} \quad \tau \boldsymbol{x} + (1 - \tau)\boldsymbol{y} \in \mathcal{F} \text{ for some } \tau \in (0, 1) \quad \text{imply} \quad \boldsymbol{x}, \boldsymbol{y} \in \mathcal{F}.$$

That is, an average of points from $\mathcal{K}$ lies in $\mathcal{F}$ if and only if the points themselves lie in $\mathcal{F}$.

The faces of a closed convex set $\mathcal{K}$ are again closed convex sets. The 0-dimensional faces are commonly called *extreme points*, and 1-dimensional faces are *edges*. The set $\mathcal{K}$ is a face of itself with maximal dimension, while faces of $\mathcal{K}$ with one dimension fewer are called *facets*.

Faces have a number of important properties. From the definition, it is clear that the "face of" relation is transitive: if $\mathcal{F}'$ is a face of $\mathcal{F}$ and $\mathcal{F}$ is a face of $\mathcal{K}$, then $\mathcal{F}'$ is a face of $\mathcal{K}$. The next fact states that the faces of a closed convex set partition the set; see [Sch14, Thm. 2.12] for the proof.

Fact 4.2 (facial decomposition). *Let $\mathcal{K} \subseteq \mathsf{V}$ be a closed convex set. Every point in $\mathcal{K}$ is contained in the relative interior of a unique face of $\mathcal{K}$.*

4.2. Simplices. A *simplex* is the convex hull of an affinely independent point set. We frequently refer to *simplicial faces* of a convex set, by which we mean faces of the set that are also simplices. The following result gives a complete description of the faces of a simplex; see [Bar02, Chap. VI.1].

Fact 4.3 (faces of a simplex). *Let $\mathcal{P} = \operatorname{conv}\{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$ be a simplex in V. For each subset $I \subseteq \{1, \dots, N\}$, the set $\operatorname{conv}\{\boldsymbol{x}_i : i \in I\}$ is a simplicial face of $\mathcal{P}$. Moreover, every face of $\mathcal{P}$ takes this form.*

A related result holds for the simplicial faces of more general convex sets.

Lemma 4.4. *Suppose that $\mathcal{F} \subseteq \mathcal{K}$ is a simplicial face of a closed convex set $\mathcal{K}$. Then every face of $\mathcal{F}$ must also be a simplicial face of $\mathcal{K}$.*

Proof. By transitivity, a face $\mathcal{F}'$ of $\mathcal{F}$ is also a face of $\mathcal{K}$. By Fact 4.3, $\mathcal{F}'$ is a simplex. ∎

4.3. Uniqueness of convex decompositions. Simplices are closely related to the uniqueness of convex decompositions. Together, the theorems of Minkowski [Sch14, Cor. 1.4.5] and Carathéodory [Sch14, Thm. 1.1.4] ensure that every point in a compact convex set can be

written as a proper convex combination of an affinely independent family of extreme points. Each of these representations is uniquely determined (up to the ordering of the extreme points) if and only if the set is a simplex.

Lemma 4.5 (unique decomposition of all points). *Let $\mathcal{K} \subset \mathsf{V}$ be a compact convex set. Each one of the points in the relative interior of $\mathcal{K}$ enjoys a unique decomposition as a proper convex combination of extreme points of $\mathcal{K}$ if and only if $\mathcal{K}$ is a simplex.*

Proof. Lacking a good reference, we offer a proof. Assume that $\mathcal{K}$ is a simplex. Then $\mathcal{K} = \operatorname{conv} \mathcal{X}$, where $\mathcal{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\} \subset \mathsf{V}$ is an affinely independent family. Using the definition of an extreme point, it is easy to verify that the extreme points of $\mathcal{K}$ are precisely the elements of $\mathcal{X}$. Now, for any point $\boldsymbol{y}$ in the affine hull of $\mathcal{X}$, we can find a representation of $\boldsymbol{y}$ as an affine combination of the points in $\mathcal{X}$ by solving the linear system

$$\sum\nolimits_{i=1}^{N} \alpha_i \boldsymbol{x}_i = \boldsymbol{y} \quad \text{and} \quad \sum\nolimits_{i=1}^{N} \alpha_i = 1.$$

Since the family $\mathcal{X}$ is affinely independent, this linear system is nonsingular, and its solution is uniquely determined. By [Sch14, Lem. 1.1.12], the representing coefficients $\alpha_1, \dots, \alpha_N$ are positive precisely when $\boldsymbol{y}$ belongs to the relative interior of the simplex $\mathcal{K} = \operatorname{conv} \mathcal{X}$.

Conversely, assume that $\mathcal{K}$ is not a simplex. By Minkowski's theorem [Sch14, Thm. 1.4.5], we can write $\mathcal{K} = \operatorname{conv} \mathcal{X}$, where $\mathcal{X}$ is the set of extreme points of $\mathcal{K}$. Since $\mathcal{K}$ is not a simplex, $\mathcal{X}$ is not affinely independent. Radon's theorem [Sch14, Thm. 1.1.5] ensures that there are two finite, disjoint subsets of $\mathcal{X}$ whose convex hulls intersect. Each point in the intersection lacks a unique representation as a proper convex combination of extreme points of $\mathcal{K}$. ∎

We can now give a precise description of when a specific point in a polytope admits a unique decomposition. This following result is a direct consequence of Fact 4.2 and Lemma 4.5.

Proposition 4.6 (unique decomposition of one point). *Let $\mathcal{K} \subset \mathsf{V}$ be a compact convex set, and fix a point $\boldsymbol{x} \in \mathcal{K}$. Then the point $\boldsymbol{x}$ admits a unique decomposition as a proper convex combination of extreme points of $\mathcal{K}$ if and only if the point $\boldsymbol{x}$ is contained in the relative interior of a simplicial face of $\mathcal{K}$.*

Proof. Let $\boldsymbol{x} \in \mathcal{K}$. According to Fact 4.2, the point $\boldsymbol{x}$ belongs to the relative interior of a unique face $\mathcal{F}$ of $\mathcal{K}$. By Definition 4.1 of a face, the point $\boldsymbol{x}$ can be written as a proper convex combination of extreme points in $\mathcal{K}$ if and only if the participating extreme points all belong to $\mathcal{F}$. Lemma 4.5 promises that $\boldsymbol{x}$ has a unique representation as a proper convex combination of the extreme points of $\mathcal{F}$ if and only if $\mathcal{F}$ is a simplex. ∎

Proposition 3.2 is just the specialization of Proposition 4.6 to the cut polytope $\mathcal{C}_n$. Meanwhile, Corollary 3.3 is the specialization to the elliptope $\mathcal{E}_n$.

4.4. Simplicial faces of the elliptope.

As we have seen, the simplicial faces of convex bodies play a central role in determining when convex representations are unique. In this section, we begin our investigation into simplicial faces of the elliptope.

4.4.1. Schur independence.

First, recall that a set $\{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\} \subseteq \{\pm 1\}^n$ of sign vectors is Schur independent when the multiset $\{\mathbf{e}\} \cup \{\boldsymbol{s}_i \odot \boldsymbol{s}_j : 1 \le i < j \le r\} \subset \mathbb{R}^n$ is linearly independent (and, in particular, contains no duplicated vector). It is easy to check that Schur independence implies ordinary linear independence.

Lemma 4.7 (Schur independence implies linear independence). *A Schur-independent set of sign vectors is also linearly independent.*

Proof. Let $\{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\} \subseteq \{\pm 1\}^n$ be Schur independent. Suppose that $\lambda_1, \dots, \lambda_r$ are real coefficients for which $\sum_{i=1}^r \lambda_i \boldsymbol{s}_i = \mathbf{0}$. Since $\boldsymbol{s}_1 \odot \boldsymbol{s}_1 = \mathbf{e}$,

$$\mathbf{0} = \boldsymbol{s}_1 \odot \mathbf{0} = \sum\nolimits_{i=1}^r \lambda_i \boldsymbol{s}_1 \odot \boldsymbol{s}_i = \lambda_1 \mathbf{e} + \sum\nolimits_{i=2}^r \lambda_i \boldsymbol{s}_1 \odot \boldsymbol{s}_i.$$

Schur independence forces the multiset $\{\mathbf{e}\} \cup \{\boldsymbol{s}_1 \odot \boldsymbol{s}_i : 2 \le i \le r\}$ to be linearly independent. We conclude that $\lambda_1 = \cdots = \lambda_r = 0$. ∎

4.4.2. Schur independence and simplicial faces. Laurent and Poljak [LP96] identified the concept of Schur independence in their work on the structure of the elliptope. In particular, they proved that Schur independence provides a sufficient condition for rank-one sign matrices to generate a simplicial face of the elliptope. This is the implication (1) ⇒ (2) in Theorem 3.4. The following result establishes the converse direction (2) ⇒ (1), which is new.

Lemma 4.8. *Let $\mathcal{S} = \{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\} \subseteq \{\pm 1\}^n$ be a set of sign vectors. If $\operatorname{conv}\{\boldsymbol{s}\boldsymbol{s}^\mathsf{t} : \boldsymbol{s} \in \mathcal{S}\}$ is a simplicial face of the elliptope $\mathcal{E}_n$, then $\mathcal{S}$ must be Schur independent.*

Proof. Suppose that $\mathcal{F} = \operatorname{conv}\{\boldsymbol{s}\boldsymbol{s}^\mathsf{t} : \boldsymbol{s} \in \mathcal{S}\}$ is a simplicial face of $\mathcal{E}_n$. We argue by contradiction.

First, assume that the family $\mathcal{S}$ is linearly independent but not Schur independent. Then the matrix $\boldsymbol{S} = \begin{bmatrix} \boldsymbol{s}_1 & \dots & \boldsymbol{s}_r \end{bmatrix} \in \{\pm 1\}^{n \times r}$ has full column rank. Moreover, the absence of Schur independence implies that there are scalars θ_0 and $\theta_{ij} = \theta_{ji}$, not all vanishing, for which

$$\theta_0 \mathbf{e} + \sum\nolimits_{i \neq j} \theta_{ij} \boldsymbol{s}_i \odot \boldsymbol{s}_j = \mathbf{0}.$$

Define a matrix $\boldsymbol{\Theta} \in \mathbb{H}_r$ whose entries are $[\boldsymbol{\Theta}]_{ii} = 0$ for each i and $[\boldsymbol{\Theta}]_{ij} = \theta_{ij}$ for $i \neq j$. For a parameter $\varepsilon > 0$, we can introduce a pair of matrices

$$\boldsymbol{A}_\pm = \boldsymbol{S} \left(\frac{1 \pm \varepsilon\theta_0}{r} \mathbf{I} \pm \varepsilon \boldsymbol{\Theta} \right) \boldsymbol{S}^\mathsf{t} \in \mathbb{H}_n.$$

Whenever ε is sufficiently small, both matrices $\boldsymbol{A}_\pm$ are psd. Furthermore, by construction,

$$\begin{aligned} \operatorname{diag}(\boldsymbol{A}_\pm) &= \frac{1 \pm \varepsilon\theta_0}{r} \sum\nolimits_{i=1}^r \operatorname{diag}(\boldsymbol{s}_i \boldsymbol{s}_i^\mathsf{t}) \pm \varepsilon \sum_{i \neq j} \theta_{ij} \operatorname{diag}(\boldsymbol{s}_i \boldsymbol{s}_j^\mathsf{t}) \\ &= \mathbf{e} \pm \varepsilon \left(\theta_0 \mathbf{e} + \sum\nolimits_{i \neq j} \theta_{ij} \boldsymbol{s}_i \odot \boldsymbol{s}_j \right) = \mathbf{e} \pm \varepsilon \mathbf{0} = \mathbf{e}. \end{aligned}$$

In other words, both matrices $\boldsymbol{A}_\pm$ belong to the elliptope $\mathcal{E}_n$. Next, we verify that the average of the two matrices coincides with the barycenter of the set $\mathcal{F}$. That is,

$$\frac{1}{2}(\boldsymbol{A}_+ + \boldsymbol{A}_-) = \frac{1}{r} \boldsymbol{S}\mathbf{I}\boldsymbol{S}^\mathsf{t} = \frac{1}{r} \sum\nolimits_{i=1}^r \boldsymbol{s}_i \boldsymbol{s}_i^\mathsf{t} \in \mathcal{F}.$$

On the other hand, neither $\boldsymbol{A}_+$ nor $\boldsymbol{A}_-$ is contained in $\mathcal{F}$. To see why, just observe that the family $\{\boldsymbol{s}_i \boldsymbol{s}_j^\mathsf{t} : 1 \le i, j \le r\}$ is linearly independent because $\boldsymbol{S}$ has full column rank. Thus, the

nonzero off-diagonal entries in $\boldsymbol{\Theta}$ contribute to $\boldsymbol{A}_{\pm}$ a nonzero matrix that does not belong to $\mathcal{F}$. But this contradicts the defining property of a face, Definition 4.1. Indeed, $\boldsymbol{A}_{\pm} \in \mathcal{E}_n$ and $\frac{1}{2}(\boldsymbol{A}_+ + \boldsymbol{A}_-) \in \mathcal{F}$, but $\boldsymbol{A}_{\pm} \notin \mathcal{F}$.

Last, assume that the family $\mathcal{S}$ of sign vectors is neither linearly independent nor Schur independent. Let $\mathcal{S}'$ be a maximal linearly independent subset of $\mathcal{S}$. Define the set $\mathcal{F}' = \operatorname{conv}\{\boldsymbol{s}\boldsymbol{s}^{\mathsf{t}} : \boldsymbol{s} \in \mathcal{S}'\}$. Fact 4.3 implies that $\mathcal{F}'$ is a simplicial face of the simplex $\mathcal{F}$. By transitivity, $\mathcal{F}'$ is also a simplicial face of $\mathcal{E}_n$. We can now repeat the argument from the last paragraph with the simplicial face $\mathcal{F}'$ and the set $\mathcal{S}'$. Again, we reach a contradiction. ∎

4.5. Explicit separators for faces of the elliptope. In the last section, we completed a characterization of the simplicial faces of the elliptope that are generated by rank-one sign matrices. A correlation matrix $\boldsymbol{A}$ is contained in a simplicial face $\mathcal{F}$ of the elliptope if and only if its sign components are Schur independent. In this section, we show how to isolate the face $\mathcal{F}$ by intersecting the elliptope $\mathcal{E}_n$ with the level set of a simple linear functional. This linear functional only depends on the range of $\boldsymbol{A}$, and it leads to a simple semidefinite representation of the face; see Proposition 3.6.

The argument is based on another remarkable property of the elliptope. Every face $\mathcal{F}$ of $\mathcal{E}_n$ is *exposed*. That is, for each face $\mathcal{F} \subseteq \mathcal{E}_n$, there exists a supporting hyperplane $\mathcal{H}$ of $\mathcal{E}_n$ such that $\mathcal{F} = \mathcal{E}_n \cap \mathcal{H}$. This is a consequence of the following result [LP95, Prop. 2.7].

Fact 4.9 (elliptope: facial membership). *Let $\boldsymbol{A}, \boldsymbol{X} \in \mathcal{E}_n$. Let $\mathcal{F}(\boldsymbol{A})$ be the smallest face of $\mathcal{E}_n$ containing $\boldsymbol{A}$. The matrix $\boldsymbol{X} \in \mathcal{F}(\boldsymbol{A})$ if and only if $\ker(\boldsymbol{A}) \subseteq \ker(\boldsymbol{X})$.*

Using Fact 4.9, we can obtain an explicit construction of a hyperplane that isolates a face of the elliptope. Proposition 3.6 is an immediate consequence of this construction.

Corollary 4.10 (elliptope: separating faces). *Let $\mathcal{F}$ be a face of the elliptope $\mathcal{E}_n$. Select a point $\boldsymbol{A}$ in the relative interior of $\mathcal{F}$, and let $\boldsymbol{P} \in \mathbb{H}_n$ be the orthogonal projector onto the range of $\boldsymbol{A}$. Then the level set*

$$\mathcal{H} = \big\{\boldsymbol{X} \in \mathbb{H}_n : \; n^{-1}\operatorname{tr}(\boldsymbol{P}\boldsymbol{X}) = 1\big\} \subset \mathbb{H}_n$$

is a hyperplane that separates $\mathcal{F}$ from $\mathcal{E}_n$. That is, $\mathcal{F} = \mathcal{E}_n \cap \mathcal{H}$.

Proof. The matrix $\boldsymbol{A}$ is contained in the relative interior of $\mathcal{F}$, so $\mathcal{F}$ is the smallest face of the elliptope that contains $\boldsymbol{A}$. A correlation matrix $\boldsymbol{X} \in \mathcal{E}_n$ obeys $\operatorname{trace}(\boldsymbol{P}\boldsymbol{X}) = \operatorname{trace}(\boldsymbol{X}) = n$ if and only if $\operatorname{range}(\boldsymbol{X}) \subseteq \operatorname{range}(\boldsymbol{P}) = \operatorname{range}(\boldsymbol{A})$. By duality, $\operatorname{range}(\boldsymbol{X}) \subseteq \operatorname{range}(\boldsymbol{A})$ if and only if $\ker(\boldsymbol{A}) \subseteq \ker(\boldsymbol{X})$. The claim now follows from Fact 4.9: A matrix $\boldsymbol{X} \in \mathcal{E}_n$ obeys $n^{-1}\operatorname{trace}(\boldsymbol{P}\boldsymbol{X}) = 1$ if and only if $\boldsymbol{X} \in \mathcal{F}$. ∎

5. Computing a sign component decomposition. In the last section, we developed a geometric analysis of the sign component decomposition (2.1) by making a connection with simplicial faces of the elliptope. Having completed this groundwork, we can prove Theorem 3.7, which states that Algorithm 2.1 is a correct method for computing sign component decompositions.

5.1. Step 1: Random optimization. Our first goal is to justify the claim that random optimization allows us to exhibit one of the rank-one sign matrix factors in the sign component decomposition (3.5) of the matrix $\boldsymbol{A}$. We derive this conclusion from a more general result.

Lemma 5.1 (random optimization). *Consider a family $\mathcal{U} = \{\boldsymbol{u}_1, \dots, \boldsymbol{u}_r\} \subset \mathbb{R}^n$, in which no pair of vectors satisfies $\boldsymbol{u}_i = \pm \boldsymbol{u}_j$ when $i \neq j$. Introduce a convex set of symmetric matrices*

$$\mathcal{P} = \operatorname{conv}\{\boldsymbol{u}\boldsymbol{u}^{\mathsf{t}} : \boldsymbol{u} \in \mathcal{U}\} \subset \mathbb{H}_n.$$

Draw a standard normal vector $\boldsymbol{g} \in \mathbb{R}^n$, and construct the linear functional $f(\boldsymbol{X}) = \boldsymbol{g}^{\mathsf{t}}\boldsymbol{X}\boldsymbol{g}$ for $\boldsymbol{X} \in \mathbb{H}_n$. Then, with probability one, there exists an index $1 \le k \le r$ for which

$$f(\boldsymbol{u}_k\boldsymbol{u}_k^{\mathsf{t}}) > f(\boldsymbol{X}) \quad \text{for all } \boldsymbol{X} \in \mathcal{P}.$$

Proof. Since $\mathcal{U}$ is finite, the maximum value of f over the convex hull $\mathcal{P}$ satisfies

$$\max_{\boldsymbol{X} \in \mathcal{P}} f(\boldsymbol{X}) = \max_{\boldsymbol{\alpha} \in \Delta_r} \sum\nolimits_{i=1}^{r} \alpha_i f(\boldsymbol{u}_i\boldsymbol{u}_i^{\mathsf{t}}) = \max_{1 \le i \le r} f(\boldsymbol{u}_i\boldsymbol{u}_i^{\mathsf{t}}).$$

Moreover, if $f(\boldsymbol{u}_k\boldsymbol{u}_k^{\mathsf{t}}) > f(\boldsymbol{u}_i\boldsymbol{u}_i^{\mathsf{t}})$ for all $i \neq k$, then the maximum on the left-hand side is attained uniquely at the matrix $\boldsymbol{X} = \boldsymbol{u}_k\boldsymbol{u}_k^{\mathsf{t}}$.

It suffices to prove that, with probability one, the linear functional f takes distinct values at the rank-one matrices $\boldsymbol{u}\boldsymbol{u}^{\mathsf{t}}$ given by $\boldsymbol{u} \in \mathcal{U}$. First, observe that $f(\boldsymbol{u}\boldsymbol{u}^{\mathsf{t}}) = \langle \boldsymbol{g},\ \boldsymbol{u}\rangle^2$. A short calculation reveals that

$$f(\boldsymbol{u}\boldsymbol{u}^{\mathsf{t}}) = f(\boldsymbol{v}\boldsymbol{v}^{\mathsf{t}}) \quad \text{if and only if} \quad \langle \boldsymbol{g},\ \boldsymbol{u} + \boldsymbol{v}\rangle = 0 \quad \text{or} \quad \langle \boldsymbol{g},\ \boldsymbol{u} - \boldsymbol{v}\rangle = 0.$$

By rotational invariance, each of the inner products follows a normal distribution:

$$\langle \boldsymbol{g},\ \boldsymbol{u} + \boldsymbol{v}\rangle \sim \textsc{normal}\big(0, \|\boldsymbol{u} + \boldsymbol{v}\|_{\ell_2}^2\big) \quad \text{and} \quad \langle \boldsymbol{g},\ \boldsymbol{u} - \boldsymbol{v}\rangle \sim \textsc{normal}\big(0, \|\boldsymbol{u} - \boldsymbol{v}\|_{\ell_2}^2\big).$$

Unless $\boldsymbol{v} = \pm\boldsymbol{u}$, neither variance can vanish. As a consequence, we may evaluate the probability

$$\begin{aligned}\mathbb{P}\left\{\langle \boldsymbol{g},\ \boldsymbol{u}_i\rangle^2 = \langle \boldsymbol{g},\ \boldsymbol{u}_j\rangle^2 \text{ for some } i \neq j\right\} \\ \le \sum\nolimits_{i<j} \left(\mathbb{P}\left\{\langle \boldsymbol{g},\ \boldsymbol{u}_i + \boldsymbol{u}_j\rangle = 0\right\} + \mathbb{P}\left\{\langle \boldsymbol{g},\ \boldsymbol{u}_i - \boldsymbol{u}_j\rangle = 0\right\}\right) = 0.\end{aligned}$$

The last relation holds because $\boldsymbol{u}_i$ never coincides with $\pm\boldsymbol{u}_j$ for $i < j$. Take the complement of this event to reach the conclusion. ∎

5.2. Step 2: Deflation. Random optimization allows us to identify a single sign component in the decomposition (3.5). In order to iterate, we must remove the contribution of this sign component from the matrix that we are factoring. The following general result shows how to extract a rank-one factor from a psd matrix, leaving a convex combination of the other rank-one factors.

Lemma 5.2 (deflation). *Consider a linearly independent family $\mathcal{U} = \{\boldsymbol{u}_1, \dots, \boldsymbol{u}_r\} \subset \mathbb{R}^n$, and suppose that*

$$\boldsymbol{M} = \sum\nolimits_{i=1}^{r} \alpha_i \boldsymbol{u}_i\boldsymbol{u}_i^{\mathsf{t}}, \quad \text{where} \quad \boldsymbol{u}_i \in \mathcal{U} \quad \text{and} \quad (\alpha_1, \dots, \alpha_r) \in \Delta_r^+.$$

Fix an index $1 \le k \le r$, and consider the semidefinite program

$$\operatorname*{maximize}_{\zeta \in \mathbb{R}} \quad \zeta \quad \text{subject to} \quad \zeta \boldsymbol{M} + (1-\zeta)\boldsymbol{u}_k\boldsymbol{u}_k^{\mathsf{t}} \succcurlyeq \mathbf{0}.$$

For the unique solution $\zeta_\star = (1-\alpha_k)^{-1}$, it holds that

$$\boldsymbol{M}' = \zeta_\star \boldsymbol{M} + (1-\zeta_\star)\boldsymbol{u}_k\boldsymbol{u}_k^{\mathsf{t}} = \sum\nolimits_{i \neq k} \frac{\alpha_i}{1-\alpha_k}\boldsymbol{u}_i\boldsymbol{u}_i^{\mathsf{t}} \in \operatorname{relint}\operatorname{conv}\left\{\boldsymbol{u}_i\boldsymbol{u}_i^{\mathsf{t}} : 1 \le i \le r \text{ and } i \neq k\right\}.$$

Proof. Without loss of generality, assume that $k = 1$. Since $\mathcal{U}$ is linearly independent, the matrix $\boldsymbol{U} = \begin{bmatrix}\boldsymbol{u}_1 & \dots & \boldsymbol{u}_r\end{bmatrix}$ has full column rank. In turn, the conjugation rule (Fact 1.1) implies that

$$\zeta \boldsymbol{M} + (1-\zeta)\boldsymbol{u}_1\boldsymbol{u}_1^{\mathsf{t}} = \boldsymbol{U}\ \operatorname{diag}\left(\alpha_1\zeta + (1-\zeta), \alpha_2\zeta, \dots, \alpha_r\zeta\right)\boldsymbol{U}^{\mathsf{t}} \succcurlyeq \mathbf{0}$$

if and only if the diagonal matrix is psd. Equivalently, ζ is feasible if and only if $0 \le \zeta \le (1-\alpha_1)^{-1}$. The optimal point $\zeta_\star$ for the semidefinite program saturates the upper bound. The second claim follows readily from a direct computation. ∎

5.3. Proof of Theorem 3.7. We are now prepared to prove Theorem 3.7, which states that Algorithm 2.1 is correct. The argument uses induction on the rank of the input matrix.

First, suppose that $\boldsymbol{A} = \boldsymbol{s}\boldsymbol{s}^{\mathsf{t}}$ is a rank-one correlation matrix generated by a sign vector $\boldsymbol{s} \in \{\pm 1\}^n$. In this case, the sign component decomposition of $\boldsymbol{A}$ is already manifest. By factorizing $\boldsymbol{A}$, we obtain the computed sign component $\pm\boldsymbol{s}$.

Now, for $r \ge 2$, suppose that $\boldsymbol{A}$ is a rank-r correlation matrix with sign component decomposition

$$\boldsymbol{A} = \sum\nolimits_{i=1}^{r} \tau_i \boldsymbol{s}_i\boldsymbol{s}_i^{\mathsf{t}} \quad \text{for} \quad \boldsymbol{s}_i \in \{\pm 1\}^n \quad \text{and} \quad (\tau_1, \dots, \tau_r) \in \Delta_r^+. \tag{5.1}$$

We assume that $\mathcal{S} = \{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\}$ is Schur independent. Lemma 4.7 states that $\mathcal{S}$ is linearly independent. In particular, $\boldsymbol{s}_i \neq \pm\boldsymbol{s}_j$ when $i \neq j$. Theorem 3.4 implies that $\mathcal{F} = \{\boldsymbol{s}\boldsymbol{s}^{\mathsf{t}} : \boldsymbol{s} \in \mathcal{S}\}$ is a simplicial face of the elliptope that contains the matrix $\boldsymbol{A}$ in its relative interior.

Compute the orthogonal projector $\boldsymbol{P} \in \mathbb{H}_n$ onto $\operatorname{range}(\boldsymbol{A})$, and define the linear functional $\psi(\boldsymbol{X}) = n^{-1}\operatorname{trace}(\boldsymbol{P}\boldsymbol{X})$ that exposes the face $\mathcal{F}$. Draw a standard normal vector $\boldsymbol{g} \in \mathbb{R}^n$. Find the solution $\boldsymbol{X}_\star$ to the semidefinite program

$$\operatorname*{maximize}_{\boldsymbol{X} \in \mathbb{H}_n} \quad \boldsymbol{g}^{\mathsf{t}}\boldsymbol{X}\boldsymbol{g} \quad \text{subject to} \quad \psi(\boldsymbol{X}) = 1 \quad \text{and} \quad \boldsymbol{X} \in \mathcal{E}_n. \tag{5.2}$$

According to Proposition 3.6, the feasible set of this optimization problem is precisely the simplicial face $\mathcal{F}$ that contains $\boldsymbol{A}$. An application of Lemma 5.1 shows that the optimal point is unique with probability one, and $\boldsymbol{X}_\star = \boldsymbol{s}_k\boldsymbol{s}_k^{\mathsf{t}}$ for some index $1 \le k \le r$. By factorizing $\boldsymbol{X}_\star$, we compute one sign component $\pm\boldsymbol{s}_k$. This justifies step 1 of Algorithm 2.1.

Next, we find the unique solution $\zeta_\star$ to the semidefinite program

$$\operatorname*{maximize}_{\zeta \in \mathbb{R}} \quad \zeta \quad \text{subject to} \quad \zeta \boldsymbol{A} + (1-\zeta)\boldsymbol{X}_\star \succcurlyeq \mathbf{0}. \tag{5.3}$$

Lemma 5.2 shows that $\zeta_\star = (1 - \tau_k)^{-1}$, where τ_k is the coefficient associated with $\boldsymbol{s}_k \boldsymbol{s}_k^\mathsf{t}$ in the representation (5.1) of the matrix $\boldsymbol{A}$. Moreover, we can form the matrix

$$\boldsymbol{A}' = \zeta_\star \boldsymbol{A} + (1 - \zeta_\star)\boldsymbol{X}_\star = \sum\nolimits_{i \neq k} \frac{\tau_i}{1 - \tau_k} \boldsymbol{s}_i \boldsymbol{s}_i^\mathsf{t} =: \sum\nolimits_{i \neq k} \tau_i' \boldsymbol{s}_i \boldsymbol{s}_i^\mathsf{t}, \quad \text{where} \quad \boldsymbol{\tau}' \in \Delta_{r-1}^+. \tag{5.4}$$

Recall that every subset of a Schur independent set remains Schur independent. Therefore, step 2 of Algorithm 2.1 produces a correlation matrix $\boldsymbol{A}'$ with rank $r - 1$ that admits a sign component decomposition (5.4) whose sign components form a Schur-independent family.

By induction, we can apply the same procedure to compute the sign components of the matrix $\boldsymbol{A}'$ defined in (5.4). This justifies the iteration procedure, step 3 in Algorithm 2.1.

Last, suppose that $\{\tilde{\boldsymbol{s}}_1, \dots, \tilde{\boldsymbol{s}}_r\} \subseteq \{\pm 1\}^n$ is the set of sign components computed by this iteration. There is a permutation π such that $\boldsymbol{s}_{\pi(i)} = \xi_i \tilde{\boldsymbol{s}}_i$ and $\xi_i \in \{\pm 1\}$ for each index $i = 1, \dots, r$. To determine the convex coefficients in the sign component decomposition of $\boldsymbol{A}$, we find the solution $\tilde{\boldsymbol{\tau}} \in \mathbb{R}^r$ to the linear system

$$\boldsymbol{A} = \sum\nolimits_{i=1}^{r} \tilde{\tau}_i \tilde{\boldsymbol{s}}_i \tilde{\boldsymbol{s}}_i^\mathsf{t}.$$

The computed sign components must be linearly independent (since the original sign components are linearly independent), so the linear system has a unique solution. In view of (5.1), it must be the case that $\tau_{\pi(i)} = \tilde{\tau}_i$ for each index i. In other words, $\{(\tilde{\tau}_i, \tilde{\boldsymbol{s}}_i) : i = 1, \dots, r\}$ is a parametric representation of the sign component decomposition of $\boldsymbol{A}$. This justifies step 4 of Algorithm 2.1, and the proof is complete.

Remark 5.3 (accuracy). Since the sign components are discrete, we can identify each one by solving the random optimization problem (5.2) with rather limited accuracy. In contrast, to remove the sign component completely, we should solve the deflation problem (5.3) to high accuracy. The deflation step (5.3) can be rewritten as a generalized eigenvalue problem, which makes this task routine.

6. Binary component decomposition.

In this section, we develop a procedure (Algorithm 2.2) for binary component decomposition, and we prove that it succeeds under a Schur independence condition (Theorem II). Our approach reduces the problem of computing a binary component decomposition to the problem of computing a sign component decomposition.

6.1. Correspondence between binary vectors and sign vectors.

Recall that we can place sign vectors and binary vectors in one-to-one correspondence via the affine map

$$\boldsymbol{F} : \{0, 1\}^n \to \{\pm 1\}^n, \quad \text{where} \quad \boldsymbol{F} : \boldsymbol{z} \mapsto 2\boldsymbol{z} - \mathbf{e} \quad \text{and} \quad \boldsymbol{F}^{-1} : \boldsymbol{s} \mapsto \tfrac{1}{2}(\boldsymbol{s} + \mathbf{e}).$$

The correspondence between sign component decompositions and binary component decompositions, however, is more subtle because they are invariant under different symmetries. Indeed, $\boldsymbol{s}\boldsymbol{s}^\mathsf{t}$ is invariant under flipping the sign of $\boldsymbol{s} \in \{\pm 1\}^n$, while $\boldsymbol{z}\boldsymbol{z}^\mathsf{t}$ is uniquely determined for each $\boldsymbol{z} \in \{0, 1\}^n$.

6.2. Reducing binary component decomposition to sign component decomposition.

Given a matrix that has a binary component decomposition, we can solve a linear system to obtain a matrix that has a closely related sign component decomposition.

Proposition 6.1 (binary component decomposition: reduction). *Consider a matrix* $\boldsymbol{H} \in \mathbb{H}_n$ *that has a binary component decomposition*

$$\boldsymbol{H} = \sum\nolimits_{i=1}^{r} \tau_i \boldsymbol{z}_i \boldsymbol{z}_i^{\mathsf{t}} \quad \text{for} \quad \boldsymbol{z}_i \in \{0,1\}^n \quad \text{and} \quad (\tau_1, \dots, \tau_r) \in \Delta_r^+. \tag{6.1}$$

Define the correlation matrix $\boldsymbol{A} \in \mathcal{E}_n$ *with sign component decomposition*

$$\boldsymbol{A} = \sum\nolimits_{i=1}^{r} \tau_i \boldsymbol{s}_i \boldsymbol{s}_i^{\mathsf{t}}, \quad \text{where} \quad \boldsymbol{s}_i = \boldsymbol{F}(\boldsymbol{z}_i) \text{ for each } i.$$

Then $\boldsymbol{A}$ *is the unique solution to the linear system*

$$\operatorname{diag}(\boldsymbol{X}) = \mathbf{e} \quad \text{and} \quad \boldsymbol{R}(4\boldsymbol{H} - \boldsymbol{X})\boldsymbol{R} = \mathbf{0}, \quad \text{where } \boldsymbol{X} \in \mathbb{H}_n. \tag{6.2}$$

Here, $\boldsymbol{R} = \mathbf{I} - n^{-1}\mathbf{e}\mathbf{e}^{\mathsf{t}}$ *denotes the orthogonal projector onto* $\{\mathbf{e}\}^{\perp} \subset \mathbb{R}^n$.

Proof. For a binary vector $\boldsymbol{z} \in \{0,1\}^n$, the sign vector $\boldsymbol{s} = \boldsymbol{F}(\boldsymbol{z})$ satisfies the identity $\boldsymbol{s}\boldsymbol{s}^{\mathsf{t}} = (2\boldsymbol{z} - \mathbf{e})(2\boldsymbol{z} - \mathbf{e})^{\mathsf{t}}$. The projector $\boldsymbol{R}$ annihilates the vector $\mathbf{e}$, so we can conjugate by $\boldsymbol{R}$ to obtain $\boldsymbol{R}\boldsymbol{s}\boldsymbol{s}^{\mathsf{t}}\boldsymbol{R} = 4\boldsymbol{R}\boldsymbol{z}\boldsymbol{z}^{\mathsf{t}}\boldsymbol{R}$. Instantiate this relation for each of the vectors $\boldsymbol{z}_i$ that appears in the binary component decomposition (6.1), and average using the weights $(\tau_1, \dots, \tau_r) \in \Delta_r^+$ to arrive at

$$\boldsymbol{R}\boldsymbol{A}\boldsymbol{R} = \sum\nolimits_{i=1}^{n} \tau_i \boldsymbol{R}\boldsymbol{s}_i\boldsymbol{s}_i^{\mathsf{t}}\boldsymbol{R} = 4\sum\nolimits_{i=1}^{n} \tau_i \boldsymbol{R}\boldsymbol{z}_i\boldsymbol{z}_i^{\mathsf{t}}\boldsymbol{R} = 4\boldsymbol{R}\boldsymbol{H}\boldsymbol{R}.$$

The correlation matrix $\boldsymbol{A}$ has a unit diagonal, so it solves the linear system (6.2).

We need to confirm that $\boldsymbol{A}$ is the only solution to (6.2). The kernel of the linear map $\boldsymbol{X} \mapsto \boldsymbol{R}\boldsymbol{X}\boldsymbol{R}$ on $\mathbb{H}_n$ consists of matrices with the form $\mathbf{e}\boldsymbol{x}^{\mathsf{t}} + \boldsymbol{x}\mathbf{e}^{\mathsf{t}}$ for $\boldsymbol{x} \in \mathbb{R}^n$. Therefore, we can parameterize each solution $\boldsymbol{X}$ of the second equation in (6.2) as $\boldsymbol{X} = \boldsymbol{A} + \mathbf{e}\boldsymbol{x}^{\mathsf{t}} + \boldsymbol{x}\mathbf{e}^{\mathsf{t}}$. But the first equation in (6.2) requires that

$$\mathbf{e} = \operatorname{diag}(\boldsymbol{X}) = \operatorname{diag}(\boldsymbol{A}) + \operatorname{diag}(\mathbf{e}\boldsymbol{x}^{\mathsf{t}}) + \operatorname{diag}(\boldsymbol{x}\mathbf{e}^{\mathsf{t}}) = \mathbf{e} + 2\boldsymbol{x}.$$

Therefore, $\boldsymbol{x} = \mathbf{0}$, and so $\boldsymbol{A}$ is the only matrix that solves both equations. ∎

6.3. Resolving the sign ambiguity. Proposition 6.1 shows that we can replace the matrix $\boldsymbol{H}$ by a correlation matrix $\boldsymbol{A}$ whose sign components are related to the binary components in $\boldsymbol{H}$. Let us explain how to resolve the sign ambiguity in the sign components of $\boldsymbol{A}$ to identify the correct binary components for $\boldsymbol{H}$.

Proposition 6.2 (sign ambiguity). *Instate the notation of Proposition 6.1. Assume that the correlation matrix* $\boldsymbol{A}$ *has a unique sign component decomposition with parametric representation* $\{(\tau_i, \tilde{\boldsymbol{s}}_i) : i = 1, \dots, r\}$, *and assume that the sign components form a linearly independent family. Find the unique solution* $\boldsymbol{\xi} \in \mathbb{R}^r$ *to the linear system*

$$n\sum\nolimits_{i=1}^{r} \tau_i \xi_i \tilde{\boldsymbol{s}}_i = (4\boldsymbol{H} - \boldsymbol{A})\mathbf{e} - 2n \operatorname{trace}(\boldsymbol{H})\mathbf{e}.$$

Then the binary components of $\boldsymbol{H}$ *are given by* $\boldsymbol{z}_i = \frac{1}{2}(\xi_i \tilde{\boldsymbol{s}}_i + \mathbf{e})$ *for each* $1 \le i \le r$.

Proof. For a binary vector $\boldsymbol{z} \in \{0,1\}^n$, define $\boldsymbol{s} = \boldsymbol{F}(\boldsymbol{z}) = 2\boldsymbol{z} - \mathbf{e}$. By direct computation,

$$\boldsymbol{s}\mathbf{e}^{\mathsf{t}} = 4\boldsymbol{z}\boldsymbol{z}^{\mathsf{t}} - 2\mathbf{e}\boldsymbol{z}^{\mathsf{t}} - \boldsymbol{s}\boldsymbol{s}^{\mathsf{t}}.$$

Right-multiply the last display by the vector $\mathbf{e}$ to arrive at

$$n\boldsymbol{s} = 4\boldsymbol{z}\boldsymbol{z}^{\mathsf{t}}\mathbf{e} - 2n(\boldsymbol{z}^{\mathsf{t}}\mathbf{e})\mathbf{e} - \boldsymbol{s}\boldsymbol{s}^{\mathsf{t}}\mathbf{e} = 4\boldsymbol{z}\boldsymbol{z}^{\mathsf{t}}\mathbf{e} - 2n\operatorname{trace}(\boldsymbol{z}\boldsymbol{z}^{\mathsf{t}})\mathbf{e} - \boldsymbol{s}\boldsymbol{s}^{\mathsf{t}}\mathbf{e}.$$

The last relation holds because an 0–1 vector $\boldsymbol{z}$ satisfies $\boldsymbol{z}^{\mathsf{t}}\mathbf{e} = \operatorname{trace}(\boldsymbol{z}\boldsymbol{z}^{\mathsf{t}})$. Instantiate the last display for the vectors $\boldsymbol{s}_i = \boldsymbol{F}(\boldsymbol{z}_i)$, and form the average using the weights $(\tau_1, \dots, \tau_r) \in \Delta_r^+$ to obtain

$$n\sum\nolimits_{i=1}^{r} \tau_i \boldsymbol{s}_i = 4\boldsymbol{H}\mathbf{e} - 2n\operatorname{trace}(\boldsymbol{H})\mathbf{e} - \boldsymbol{A}\mathbf{e}.$$

We have used the definitions of $\boldsymbol{H}$ and $\boldsymbol{A}$ from the statement of Proposition 6.1.

By uniqueness, the sign components $\tilde{\boldsymbol{s}}_i$ in the parametric representation coincide with the vectors $\boldsymbol{s}_i$ up to global sign flips; that is, $\boldsymbol{s}_i = \xi_i \tilde{\boldsymbol{s}}_i$, where $\xi_i \in \{\pm 1\}$ for each index i. Substitute this relation into the last display to obtain

$$n\sum\nolimits_{i=1}^{r} \tau_i \xi_i \tilde{\boldsymbol{s}}_i = 4\boldsymbol{H}\mathbf{e} - 2n\operatorname{trace}(\boldsymbol{H})\mathbf{e} - \boldsymbol{A}\mathbf{e}.$$

This is a consistent linear system in the variables $\xi_1, \dots, \xi_r$. The solution is unique because $\{\tilde{\boldsymbol{s}}_1, \dots, \tilde{\boldsymbol{s}}_r\}$ is a linearly independent family. Therefore, we can obtain the sign pattern by solving the linear system, and

$$\boldsymbol{z}_i = \boldsymbol{F}^{-1}(\boldsymbol{s}_i) = \boldsymbol{F}^{-1}(\xi_i \tilde{\boldsymbol{s}}_i) = \tfrac{1}{2}(\xi_i \tilde{\boldsymbol{s}}_i + \mathbf{e}).$$

This observation completes the argument. ∎

6.4. Computation of the binary component decomposition. Propositions 6.1 and 6.2 give us a mechanism for computing a binary component decomposition, provided that an associated matrix has a unique sign component decomposition with linearly independent sign components. We can exploit our theory on the tractable computation of sign component decompositions to identify situations where we can compute binary component decompositions.

Proposition 6.3 (Schur independence: equivalence). *A family $\{\boldsymbol{z}_1, \dots, \boldsymbol{z}_r\} \subseteq \{0,1\}^n$ of binary vectors is Schur independent if and only if the associated family $\{\mathbf{e}, \boldsymbol{F}(\boldsymbol{z}_1), \dots, \boldsymbol{F}(\boldsymbol{z}_r)\} \subseteq \{\pm 1\}^n$ of sign vectors is Schur independent.*

Proof. Set $\boldsymbol{z}_0 = \mathbf{e}$ and $\boldsymbol{s}_0 = \mathbf{e}$. For $1 \le i \le r$, define $\boldsymbol{s}_i = \boldsymbol{F}(\boldsymbol{z}_i) = 2\boldsymbol{z}_i - \mathbf{e}$. Then

$$\operatorname{span}\{\boldsymbol{z}_i \odot \boldsymbol{z}_j : 0 \le i, j \le r\} = \operatorname{span}\{\boldsymbol{s}_i \odot \boldsymbol{s}_j : 0 \le i, j \le r\}.$$

This point follows easily from the definition of the linear hull. ∎

With this result at hand, we can prove Theorem II.

Proof of Theorem II. Suppose that $\boldsymbol{H} \in \mathbb{H}_n$ has a binary component decomposition $\boldsymbol{H} = \sum_{i=1}^{r} \tau_i \boldsymbol{z}_i \boldsymbol{z}_i^{\mathsf{t}}$ involving a Schur independent family $\{\boldsymbol{z}_1, \dots, \boldsymbol{z}_r\}$ of binary components. Introduce the associated sign vectors $\boldsymbol{s}_i = \boldsymbol{F}(\boldsymbol{z}_i)$. By Proposition 6.3, the family $\{\boldsymbol{s}_1, \dots, \boldsymbol{s}_r\}$ of sign vectors is Schur independent, hence linearly independent by Lemma 4.7.

Proposition 6.1 shows that we can form the correlation matrix $\boldsymbol{A} = \sum_{i=1}^{r} \tau_i \boldsymbol{s}_i \boldsymbol{s}_i^{\mathsf{t}}$ by solving a linear system. By Theorem 3.7, Algorithm 2.1 allows us to compute pairs $(\tilde{\tau}_i, \tilde{\boldsymbol{s}}_i)$ with the property that $\tau_{\pi(i)} = \tilde{\tau}_i$ and $\boldsymbol{s}_{\pi(i)} = \xi_i \tilde{\boldsymbol{s}}_i$, where π is a permutation and $\xi_i \in \{\pm 1\}$ for each i. Proposition 6.2 shows that we can use the computed sign components to find the associated binary components $\boldsymbol{z}_{\pi(i)}$ that participate in $\boldsymbol{H}$. ∎

7. Application: Activity detection in massive MIMO systems. In this section, we outline a stylized application of binary component decomposition in modern communications.

7.1. Motivation. Massive connectivity is predicted to be a key feature in future wireless cellular networks (IoT) and Device-to-Device communication (D2D). Base stations will face the challenge of connecting a large number of devices and distributing communication resources accordingly. While this seems daunting in general, a key feature of these systems is parsimony. Individual device activity is typically sporadic. This feature can be exploited by a two-phase approach.

1. *Activity detection:* identify the (small) set of active users at a given time.
2. *Scheduling:* distribute communication resources among these active users.

Recent works have pointed out that multiple antennas at the base station may help to tackle the first phase [LY18, CSY18, HJC18]. The mathematical motivation behind this approach is covariance estimation. Massive MIMO systems allow for estimating the covariance matrix of an incoming signal, rather than the signal itself. As detailed below, this reduces the task of identifying active devices to a matrix factorization problem. The covariance matrix is proportional to a convex combination of structured rank-one factors. Each of these factors is in one-to-one correspondence with a single active device. Identifying this factorization in turn allows for solving the activity detection problem.

7.2. Signal model. Suppose that a network contains N different devices and a single base station. The base station contains M different antennas. Each of these antennas is capable of resolving n-dimensional signals. To perform activity detection, unique pilot sequences are distributed among the devices. Denote them by $\boldsymbol{a}_1, \dots, \boldsymbol{a}_N \in \mathbb{C}^n$. If device k wants to indicate activity, it transmits its pilot $\boldsymbol{a}_k$ over the shared network. At a given time, the base station receives noisy superpositions of several pilot sequences that passed through wireless channels. The channel connecting device k $(1 \le k \le N)$ with the ith antenna $(1 \le i \le M)$ is modeled by a large-scale fading coefficient $\tau_k > 0$ that is constant over all antennas and a channel vector $\bar{\boldsymbol{h}}_k \in \mathbb{C}^M$ that subsumes fluctuations between antennas:

$$\boldsymbol{y}_i = \sum\nolimits_{k \in \mathcal{A}} \sqrt{\tau_k} \left[\bar{\boldsymbol{h}}_k\right]_i \boldsymbol{a}_k + \boldsymbol{\varepsilon}_i \in \mathbb{C}^n \quad \text{for} \quad 1 \le i \le M.$$

The set $\mathcal{A} \subset \{1, \dots, N\}$ denotes the subset of active devices and $\boldsymbol{\varepsilon}_i \in \mathbb{C}^m$ represents additive noise corruption affecting the ith antenna. Simplifying assumptions, such as *white noise corruption* (each $\boldsymbol{\varepsilon}_i$ is a complex standard Gaussian vector with variance ε) and *spatially white channel vectors* (each $\boldsymbol{h}_k$ contains i.i.d. standard normal entries), imply the following simple formula for the covariance:

$$\operatorname{Cov}(\boldsymbol{y}_i) = \mathbb{E}\left[\boldsymbol{y}\boldsymbol{y}^*\right] = \sum\nolimits_{k \in \mathcal{A}} \tau_k \boldsymbol{a}_k \boldsymbol{a}_k^* + \varepsilon \mathbf{I} \quad \text{for all} \quad 1 \le i \le M.$$

The MIMO setup allows for empirically approximating this covariance:

$$\boldsymbol{Y} = M^{-1} \sum\nolimits_{i=1}^{M} \boldsymbol{y}_i \boldsymbol{y}_i^* \overset{M\to\infty}{\longrightarrow} \sum\nolimits_{k\in\mathcal{A}} \tau_k \boldsymbol{a}_k \boldsymbol{a}_k^* + \varepsilon \mathbf{I}. \tag{7.1}$$

The (quick) rate of convergence can be controlled using matrix-valued concentration inequalities [Tro12]. We refer to [HJC18] for a more detailed analysis and justification of the simplifying assumptions.

7.3. Compressed activity detection via sign component decomposition. Let N be the total number of devices in the network. Set the internal dimension to $n = \lceil \log_2(N) \rceil + 1$. Equip each device with a unique pilot sequence $\boldsymbol{a}_i = \boldsymbol{s}_i \in \{\pm 1\}^n$ such that $\boldsymbol{s}_i \neq \pm \boldsymbol{s}_j$ for all $i \neq j$. Next, assume that the base station contains sufficiently many antennas to accurately estimate the signal covariance matrix (7.1) at any given time:

$$\boldsymbol{Y} = \sum\nolimits_{k\in\mathcal{A}} \tau_k \boldsymbol{s}_k \boldsymbol{s}_k^T + \varepsilon \mathbf{I}.$$

Standard techniques let us remove the isotropic noise distortion $\varepsilon\mathbf{I}$. The remainder is proportional to a correlation matrix: $\bar{\boldsymbol{Y}} = \sum_{k\in\mathcal{A}} \tau_k \boldsymbol{s}_k \boldsymbol{s}_k^T$. The activity pattern $\mathcal{A}$ is encoded in the sign components (pilots) of this correlation matrix. Apply Algorithm 2.1 to identify them.

Theorem I asserts that this identification succeeds, provided that the participating sign components are Schur independent. This assumption imposes stringent constraints on the maximum number of active devices that can be resolved correctly; see (2.3). But beneath this threshold, Schur independence is generic. Fact 2.2 asserts that almost all activity patterns produce Schur independent pilot sequences.

The method proposed here is conceptually different from existing approaches. These assign random pilot sequences and exploit sparsity in the activity pattern (viewed as a binary vector in $\mathbb{R}^N$) either via approximate message passing [CSY18] or ideas from compressed sensing [HJC18, FJ19]. In contrast, activity detection via sign component decomposition assigns deterministic pilot sequences that are guaranteed to work for most parsimonious activity patterns. The algorithmic reconstruction cost scales polynomially in $n \simeq \log(N)$, an exponential improvement over existing rigorous reconstruction techniques [FJ19].

The arguments presented here are based on several idealizations and should be viewed as a proof of concept. We intend to address concrete implementations of MIMO activity detection via sign component decomposition in future work.

8. Related work. The goal of matrix factorization is to produce a representation of a matrix $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ as a product of structured matrices. The simplest formulation expresses

$$\boldsymbol{B} = \boldsymbol{V}\boldsymbol{W}^{\mathsf{t}} + \boldsymbol{E}, \quad \text{where} \quad \boldsymbol{V} \in \mathbb{R}^{n\times r} \quad \text{and} \quad \boldsymbol{W} \in \mathbb{R}^{m\times r}. \tag{8.1}$$

The matrix $\boldsymbol{E} \in \mathbb{R}^{n\times m}$ collects the approximation error; the factorization is *exact* if $\boldsymbol{E} = \mathbf{0}$. It is also common to normalize the factors and expose the scaling by means of a separate diagonal factor:

$$\boldsymbol{B} = \boldsymbol{V}\ \mathrm{diag}(\boldsymbol{\lambda})\, \boldsymbol{W}^{\mathsf{t}} + \boldsymbol{E}, \quad \text{where} \quad \boldsymbol{\lambda} \in \mathbb{R}^r_+. \tag{8.2}$$

We can try to extract different types of structure in the matrix by placing appropriate constraints on the factors $\boldsymbol{V}$ and $\boldsymbol{W}$. The shape of the factors may vary, depending on the application.

The basic questions about matrix factorization are existence, uniqueness, robustness, and computational tractability. Surprisingly, there remain many open theoretical questions about matrix factorizations beyond the most classical examples. Furthermore, a majority of the algorithmic work consists of heuristic nonconvex optimization procedures or methods that require generative modeling assumptions. The aim of this section is to summarize the literature on discrete factorizations, as well as some general computational approaches to matrix factorization.

8.1. Integer factorizations and factorization over finite fields. In 1851, Hermite developed an integer analog of the reduced row echelon form [Her51]. For an integer matrix $\boldsymbol{B}$, the Hermite normal form is an exact factorization (8.1) where $\boldsymbol{V}$ is a square unimodular[2] integer matrix and $\boldsymbol{W}$ is a triangular integer matrix. This decomposition is a discrete analog of the QR factorization.

Similarly, the Smith normal form [Smi61] of an integer matrix $\boldsymbol{B}$ is an exact factorization (8.2) where $\boldsymbol{V}$ and $\boldsymbol{W}$ are square unimodular integer matrices, and $\boldsymbol{\lambda} \in \mathbb{N}^r$ is an integer vector with the divisibility property $\lambda_{i+1} \,|\, \lambda_i$ for each i. It is a discrete analog of the SVD.

The Hermite and Smith normal forms of an integer matrix can be computed in strongly polynomial time [KB79]. Contemporary applications include multidimensional signal processing, lattice computations, and solving Diophantine equations [Yap00].

Both decompositions can be extended to a matrix whose entries are drawn from a principal ideal domain. For example, with respect to the finite field $\mathbb{Z}_2 = \{0, 1\}$, these normal forms lead to binary factorizations of a binary matrix in binary arithmetic. Matrix factorization in binary and Boolean [Kim82] arithmetic are an active research area with applications ranging from role engineering [LVA08] to data clustering [ZLD+10].

8.2. Semidiscrete factorizations. Kolda [Kol98] coined the term *semidiscrete factorization* to describe the factorizations of the form (8.2) where the outer factors $\boldsymbol{V}, \boldsymbol{W}$ are discrete while the diagonal vector $\boldsymbol{\lambda}$ takes real values. The literature contains several instances.

8.2.1. Integer factorizations. Tropp [Tro15] proved that every positive-definite matrix $\boldsymbol{B}$ admits an exact semidiscrete factorization (8.2) where $\boldsymbol{V} = \boldsymbol{W}$ and the factor $\boldsymbol{V}$ has integer entries that are bounded in terms of the condition number and the dimension. This result has applications in probability theory [BLX18], but the proof is nonconstructive.

8.2.2. Ternary factorizations. Motivated by applications in image processing, O'Leary and Peleg [OP83a] proposed a heuristic deflation method for computing a ternary factorization (8.2) with $\boldsymbol{V} \in \{0, \pm 1\}^{n \times r}$ and $\boldsymbol{W} \in \{0, \pm 1\}^{m \times r}$. At each step, they aim to solve the integer optimization problem

$$\text{maximize} \quad \boldsymbol{x}^{\mathsf{t}} \boldsymbol{B} \boldsymbol{y} \quad \text{subject to} \quad \boldsymbol{x} \in \{0, \pm 1\}^n \quad \text{and} \quad \boldsymbol{y} \in \{0, \pm 1\}^m. \tag{8.3}$$

[2] A unimodular matrix has determinant one.

Given an approximate solution $(\boldsymbol{x}, \boldsymbol{y})$ to (8.3), they update the target matrix as

$$\boldsymbol{B} \mapsto \boldsymbol{B} - \lambda\, \boldsymbol{x}\boldsymbol{y}^{\mathsf{t}}, \quad \text{where} \quad \lambda = \boldsymbol{x}^{\mathsf{t}}\boldsymbol{B}\boldsymbol{y}/(\|\boldsymbol{x}\|\,\|\boldsymbol{y}\|). \tag{8.4}$$

This process leads directly to a factorization of the form (8.2). Since it is computationally hard to solve the optimization problem (8.3) exactly, O'Leary and Peleg resort to alternating minimization: they fix $\boldsymbol{x}$ and minimize with respect to $\boldsymbol{y}$, they fix $\boldsymbol{y}$ and minimize with respect to $\boldsymbol{x}$, and they repeat. Kolda [Kol98, Prop. 6.2] showed that these heuristics produce a sequence of improving approximations, but there is no control on the convergence rate.

8.2.3. Binary factorizations, or cut decompositions. The *cut norm* of a matrix $\boldsymbol{B}$ is the value of the modified problem (8.3) with constraints $\boldsymbol{x} \in \{0,1\}^n$ and $\boldsymbol{y} \in \{0,1\}^m$. It has applications in graph theory and theoretical algorithms. In the late 1990s, Frieze and Kannan [FK99] proposed the *cut decomposition*, an approximate semidiscrete factorization (8.2) of a general matrix $\boldsymbol{B}$, where the outer factors $\boldsymbol{V} \in \{0,1\}^{n\times r}$ and $\boldsymbol{W} \in \{0,1\}^{m\times r}$ take binary values. They developed an algorithm that gives a rigorous tradeoff between the number r of terms in the cut decomposition and the cut-norm approximation error.

Subsequently, Alon and Naor [AN06] developed another algorithm for the cut decomposition that proceeds by a rigorous process of iterative deflation. More precisely, Alon and Naor explain how to use semidefinite relaxation and rounding to obtain a pair $(\boldsymbol{x}, \boldsymbol{y})$ where the cut norm is approximately achieved. They update the matrix $\boldsymbol{B}$ via the deflation rule (8.4). They obtain bounds that improve substantially over [FK99], but their method may not produce an exact low-rank factorization even when one exists.

8.2.4. Sign and binary component decompositions. Our work introduces exact semidiscrete factorizations, where the left factor $\boldsymbol{V}$ consists of signs $\{\pm 1\}$ or binary values $\{0,1\}$. In this paper, we consider the positive-semidefinite case, where the left and right factors match: $\boldsymbol{W} = \boldsymbol{V}$. In the companion work [KT19], we consider the asymmetric case where the right factor $\boldsymbol{W}$ is arbitrary (but might be discrete). In particular, we see that the binary component decomposition (2.4) is an exact symmetric cut decomposition.

Our work gives conditions for existence, uniqueness, and computational tractability of these factorizations. Moreover, our results do not require a generative stochastic model. A serious limitation, however, is the restriction to matrices that have exact low rank. We are seeking extensions to general matrices in ongoing research.

8.3. Other approaches to structured factorization. Because of its importance in data analysis, there is an extensive literature on structured matrix factorization. Some of the most popular examples of constrained factorization are independent component analysis [Com94], nonnegative matrix approximation [PT94], sparse coding [OF96], and sparse principal component analysis [ZHT06]. Some frameworks for thinking about structured matrix factorization appear in [TB99, CDS02, Tro04, Sre04, Wit10, Jag11, Bac13, Ude15, BE16, Bru17, HV19], among other sources. In this section, we summarize some general methods that have been proposed for matrix factorization. See [Ude15, Bru17] for more complete literature reviews.

8.3.1. Deflation. A large number of authors have proposed matrix factorization techniques based on deflation [Wit10, Jag11, Bac13, Ude15, Bru17]. The basic step in these

methods (attempts) to solve a problem like

$$\text{maximize} \quad \boldsymbol{x}^{\mathsf{t}}\boldsymbol{B}\boldsymbol{y} \quad \text{subject to} \quad P_1(\boldsymbol{x}) \le c_1 \quad \text{and} \quad P_2(\boldsymbol{y}) \le c_2. \tag{8.5}$$

The functions P_1 and P_2 are (convex) regularizers that promote structure in the rank-one factor $\boldsymbol{x}\boldsymbol{y}^{\mathsf{t}}$. For example, when P_1 is the ℓ_1 norm, this formulation tends to promote sparsity in the factors. Given an approximate solution $(\boldsymbol{x}, \boldsymbol{y})$, we update the matrix via (8.4) or using the conditional gradient method [Jag11]. Other deflation techniques have been developed for the particular task of sparse principal component analysis [Mac09]. Witten [Wit10] refers to this approach as *penalized matrix decomposition*.

In practice, the most common heuristic for solving (8.5) is alternating maximization. Other nonconvex optimization algorithms, such as projected gradient, may yield better results [Ude15]. In some special cases, we can attack the problem via semidefinite relaxation [dEGJL07]. To the best of our knowledge, these methods do not offer any guarantees.

In rare cases where we can provably compute an approximate solution to (8.5), this approach leads to a rigorous tradeoff between the number of terms in the matrix approximation and the Frobenius-norm approximation error [Tem03, Jag11]. Nevertheless, even when the matrix has an exact factorization, there is no guarantee that we will find it. Furthermore, the core optimization (8.5) is usually computationally hard, so all bets are off.

8.3.2. Direct methods. Many other authors [Gor03, Tro04, Sre04, Bac13, Ude15, Bru17, HV19] have considered problem formulations that seek to compute the matrix factors directly. These approaches frame an optimization problem like

$$\underset{\boldsymbol{V}\in\mathbb{R}^{n\times r}, \boldsymbol{W}\in\mathbb{R}^{m\times r}}{\text{minimize}} \quad \ell(\boldsymbol{B}, \boldsymbol{V}\boldsymbol{W}^{\mathsf{t}}) + P_1(\boldsymbol{V}) + P_2(\boldsymbol{W}). \tag{8.6}$$

The loss function $\ell(\cdot, \cdot)$ is typically the Frobenius norm, and the functions P_1 and P_2 are (convex) penalties that promote structure on the matrix factors.

In practice, the most common heuristic for solving (8.6) is alternating minimization. Other nonconvex algorithms, such as projected gradient, can also be applied. When the inner dimension r of the factors is large enough, the optimization problem (8.6) is sometimes tractable, in spite of the apparent nonconvexity; see [Gor03, Bac13, Bru17, HV19]. Nevertheless, we must regard (8.6) as a heuristic, except in a limited set of circumstances.

8.4. Conclusions. The results presented here take the reverse point of view from most of the existing literature. We first identify a class of matrices that admits a unique discrete factorization, and we use this insight to develop a tractable algorithm that provably computes this factorization. The next step is to understand how to find an approximate discrete factorization of a noisy matrix. We believe that this kind of factorization has several applications, including activity detection in large asynchronous networks.

Acknowledgments. The authors thank Benjamin Recht for helpful conversations at an early stage of this project. Peter Jung suggested activity detection in massive MIMO system as a potential application. We also want to thank the anonymous referees for insightful comments and suggestions that helped us to further improve this presentation.

REFERENCES

[AN06] N. Alon and A. Naor, *Approximating the cut-norm via Grothendieck's inequality*, SIAM J. Comput., 35 (2006), pp. 787–803, https://doi.org/10.1137/S0097539704441629.

[Bac13] F. R. Bach, *Convex Relaxations of Structured Matrix Factorizations*, preprint https://arxiv.org/abs/1309.3117, 2013.

[Bar02] A. Barvinok, *A Course in Convexity*, Grad. Stud. Math. 54, American Mathematical Society, Providence, RI, 2002.

[BBV16] A. S. Bandeira, N. Boumal, and V. Voroninski, *On the low-rank approach for semidefinite programs arising in synchronization and community detection*, in Proceedings of the 29th Annual Conference on Learning Theory, Columbia University, New York, NY, 2016, Proceedings of Machine Learning Research 49, V. Feldman, A. Rakhlin, and O. Shamir, eds., PMLR, pp. 361–382.

[BE16] M. Basbug and B. Engelhardt, *Hierarchical compound poisson factorization*, in Proceedings of the 33rd International Conference on Machine Learning, New York, New York, 2016, Proceedings of Machine Learning Research 48, M. F. Balcan and K. Q. Weinberger, eds., PMLR, pp. 1795–1803.

[BLX18] A. D. Barbour, M. J. Luczak, and A. Xia, *Multivariate approximation in total variation,* I: *Equilibrium distributions of Markov jump processes*, Ann. Probab., 46 (2018), pp. 1351–1404.

[BM03] S. Burer and R. D. C. Monteiro, *A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization*, Math. Program., 95 (2003), pp. 329–357.

[Bru17] J. J. Bruer, *Recovering Structured Low-rank Operators using Nuclear Norms*, Ph.D. thesis, Caltech, Pasadena, CA, 2017.

[CDS02] M. Collins, S. Dasgupta, and R. E. Schapire, *A generalization of principal component analysis to the exponential family*, in Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic, 2001, pp. 617–624.

[Com94] P. Comon, *Independent component analysis, a new concept?*, Signal Process, 36 (1994), pp. 287–314.

[CSY18] Z. Chen, F. Sohrabi, and W. Yu, *Sparse activity detection for massive connectivity*, IEEE Trans. Signal Process., 66 (2018), pp. 1890–1904.

[dEGJL07] A. d'Aspremont, L. El Ghaoui, M. I. Jordan, and G. R. G. Lanckriet, *A direct formulation for sparse PCA using semidefinite programming*, SIAM Rev., 49 (2007), pp. 434–448, https://doi.org/10.1137/050645506.

[DL97] M. M. Deza and M. Laurent, *Geometry of Cuts and Metrics*, Algorithms Combin. 15, Springer-Verlag, Berlin, 1997.

[DT96] R. A. DeVore and V. N. Temlyakov, *Some remarks on greedy algorithms*, Adv. Comput. Math., 5 (1996), pp. 173–187.

[FJ19] A. Fengler and P. Jung, *On the Restricted Isometry Property of Centered Self Khatri-Rao Products*, preprint, https://arxiv.org/abs/1905.09245, 2019.

[FK99] A. Frieze and R. Kannan, *Quick approximation to matrices and applications*, Combinatorica, 19 (1999), pp. 175–220.

[GLS93] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Algorithms and Combin. 2, 2nd ed, Springer-Verlag, Berlin, 1993.

[Gor03] G. J. Gordon, *Generalized^2 linear^2 models*, in Advances in Neural Information Processing Systems 15, S. Becker, S. Thrun, and K. Obermayer, eds., MIT Press, 2003, pp. 593–600.

[Gru07] P. M. Gruber, *Convex and Discrete Geometry*, Grundlehren Math. Wiss. 336, Springer, Berlin, 2007.

[GV18] N. Gillis and S. A. Vavasis, *On the complexity of robust PCA and ℓ_1-norm low-rank matrix approximation*, Math. Oper. Res., 43 (2018), pp. 1072–1084.

[Her51] C. Hermite, *Sur l'introduction des variables continues dans la théorie des nombres*, J. Reine Angew. Math., 41 (1851), pp. 191–216.

[HJC18] S. Haghighatshoar, P. Jung, and G. Caire, *Improved scaling law for activity detection in massive MIMO systems*, in Proceedings of the 2018 IEEE International Symposium on Information Theory (ISIT), IEEE, 2018, pp. 381–385.

[HUL01] J.-B. Hiriart-Urruty and C. Lemaréchal, *Fundamentals of Convex Analysis*, Abridged version of Convex Analysis and Minimization Algorithms. I [Springer, Berlin, 1993] and II [ibid.], Grundlehren Text Editions, Springer-Verlag, Berlin, 2001.

[HV19] B. D. Haeffele and R. Vidal, *Structured low-rank matrix factorization: Global optimality, algorithms, and applications*, IEEE Trans. Pattern Anal. Mach. Intell., 42 (2020), pp. 1468–1482.

[Jag11] M. Jaggi, *Sparse Convex Optimization Methods for Machine Learning*, Ph.D. thesis, ETH Zürich, Switzerland, 2011.

[Jol02] I. T. Jolliffe, *Principal Component Analysis*, Springer Ser. Statist., 2nd ed., Springer-Verlag, New York, 2002.

[KB79] R. Kannan and A. Bachem, *Polynomial algorithms for computing the Smith and Hermite normal forms of an integer matrix*, SIAM J. Comput., 8 (1979), pp. 499–507, https://doi.org/10.1137/0208040.

[Kim82] K. H. Kim, *Boolean Matrix Theory and Applications*, Monographs and Textbooks in Pure and Applied Mathematics, 70, Marcel Dekker, New York, 1982.

[Kol98] T. G. Kolda, *Limited-memory Matrix Methods with Applications*, Ph.D. thesis, University of Michigan, Ann Arbor, MI, 1998.

[KT19] R. Kueng and J. Tropp, *Binary Component Decomposition Part* II: *The Asymmetric Case*, preprint, 2019, https://arxiv.org/abs/1907.13602.

[LP95] M. Laurent and S. Poljak, *On a positive semidefinite relaxation of the cut polytope*, Linear Algebra Appl., 223/224 (1995), pp. 439–461.

[LP96] M. Laurent and S. Poljak, *On the facial structure of the set of correlation matrices*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 530–547, https://doi.org/10.1137/0617031.

[LVA08] H. Lu, J. Vaidya, and V. Atluri, *Optimal boolean matrix decomposition: Application to role engineering*, in Proceedings of the 24th International Conference on Data Engineering, ICDE Cancún, Mexico, 2008, G. Alonso, J. A. Blakeley, and A. L. P. Chen, eds., IEEE Computer Society, 2008, pp. 297–306.

[LY18] L. Liu and W. Yu, *Massive connectivity with massive MIMO—Part* I: *Device activity detection and channel estimation*, IEEE Trans. Signal Process., 66 (2018), pp. 2933–2946.

[Mac09] L. W. Mackey, *Deflation methods for sparse PCA*, in Advances in Neural Information Processing Systems 21, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, eds., Curran Associates, Red Hook, NY, 2009, pp. 1017–1024.

[OF96] B. A. Olshausen and D. J. Field, *Emergence of simple-cell receptive field properties by learning a sparse code for natural images*, Nature, 381 (1996), pp. 607–609.

[OP83a] D. O'Leary and S. Peleg, *Digital image compression by outer product expansion*, IEEE Trans. Communications, 31 (1983), pp. 441–444.

[PT94] P. Paatero and U. Tapper, *Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values*, Environmetrics, 5 (1994), pp. 111–126.

[Roc70] R. T. Rockafellar, *Convex Analysis*, Princeton Mathematical Series 28, Princeton University Press, Princeton, NJ, 1970.

[Sch14] R. Schneider, *Convex Bodies: The Brunn-Minkowski Theory*, Encyclopedia Math. Appl. 151, second expanded ed., Cambridge University Press, Cambridge, 2014.

[Smi61] H. J. S. Smith, *On systems of linear indeterminate equations and congruences*, Philos. Trans. Royal Soc. London, 151 (1861), pp. 293–326.

[Sre04] N. Srebro, *Learning with Matrix Factorizations*, ProQuest LLC, Ann Arbor, MI, 2004, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA.

[TB99] M. E. Tipping and C. M. Bishop, *Probabilistic principal component analysis*, J. R. Stat. Soc. Ser. B Stat. Methodol., 61 (1999), pp. 611–622.

[Tem03] V. N. Temlyakov, *Nonlinear methods of approximation*, Found. Comput. Math., 3 (2003), pp. 33–107.

[Tro04] J. A. Tropp, *Topics in Sparse Approximation*, Ph.D. thesis, The University of Texas at Austin, Austin, TX, 2004.

[Tro12] J. A. Tropp, *User-friendly tail bounds for sums of random matrices*, Found. Comput. Math., 12 (2012), pp. 389–434.

[Tro15] J. A. Tropp, *Integer factorization of a positive-definite matrix*, SIAM J. Discrete Math., 29 (2015), pp. 1783–1791, https://doi.org/10.1137/15M1024718.

[Tro18] J. A. Tropp, *Simplicial faces of the set of correlation matrices*, Discrete Comput. Geom., 60 (2018), pp. 512–529.

[Ude15] M. Udell, *Generalized Low Rank Models*, Ph.D. thesis, Stanford University, Stanford, CA, 2015.

[Wit10] D. M. Witten, *A Penalized Matrix Decomposition and its Applications*, Ph.D. thesis, Stanford University, Stanford, CA, 2010.

[Yap00] C. K. Yap, *Fundamental Problems of Algorithmic Algebra*, Oxford University Press, New York, 2000.

[YTF+19] A. Yurtsever, J. A. Tropp, O. Fercoq, M. Udell, and V. Cevher, *Scalable Semidefinite Programming*, preprint, https://arxiv.org/abs/1912.02949, 2019.

[ZHT06] H. Zou, T. Hastie, and R. Tibshirani, *Sparse principal component analysis*, J. Comput. Graph. Statist., 15 (2006), pp. 265–286.

[ZLD+10] Z. Zhang, T. Li, C. H. Q. Ding, X. Ren, and X. Zhang, *Binary matrix factorization for analyzing gene expression data*, Data Min. Knowl. Discov., 20 (2010), pp. 28–52.