

Enhancing Pure-Pixel Identification Performance via Preconditioning*

Nicolas Gillis[†] and Wing-Kin Ma[‡]

Abstract. In this paper, we analyze different preconditionings designed to enhance robustness of pure-pixel search algorithms, which are used for blind hyperspectral unmixing and which are equivalent to near-separable nonnegative matrix factorization algorithms. Our analysis focuses on the successive projection algorithm (SPA), a simple, efficient, and provably robust algorithm. Recently, a provably robust preconditioning was proposed by Gillis and Vavasis [*SIAM J. Optim.*, 25 (2015), pp. 677–698] which requires the resolution of a semidefinite program (SDP). Since solving the SDP in high precisions can be time consuming, we generalize the robustness analysis to approximate solutions of the SDP showing that a high accuracy solution is not crucial for robustness, paving the way for faster preconditionings. This first contribution also allows us to provide a robustness analysis for two other preconditionings. The first one is prewhitening, which can be interpreted as an optimal solution of the same SDP with additional constraints. We analyze the robustness of prewhitening, which allows us to characterize situations in which it performs competitively with the SDP-based preconditioning. The second one is based on SPA itself and can be interpreted as an optimal solution of a relaxation of the SDP. It is extremely fast when competing with the SDP-based preconditioning on several synthetic data sets.

Key words. hyperspectral unmixing, pure-pixel search, preconditioning, prewhitening, successive projection algorithm, near-separable nonnegative matrix factorization, robustness to noise, semidefinite programming

AMS subject classifications. 15A23, 65F30

DOI. 10.1137/140994915

1. Introduction. In this paper, we analyze a blind unmixing problem referred to as near-separable nonnegative matrix factorization (NMF). Our main motivation comes from blind hyperspectral unmixing (blind HU), which aims at recovering the spectral signatures of the constitutive materials present in a hyperspectral image, called endmembers, along with their abundances in each pixel. Under the *linear mixing model*, the spectral signature of a pixel is equal to a linear combination of the spectral signatures of the endmembers where the weights correspond to the abundances. More formally, letting $X \in \mathbb{R}_+^{m \times n}$ represent a hyperspectral image with m wavelengths and n pixels, we have, in the noiseless case,

$$X(:, j) = \sum_{k=1}^r W(:, k)H(k, j) \quad \forall j,$$

*Received by the editors November 7, 2014; accepted for publication (in revised form) March 18, 2015; published electronically May 21, 2015.

<http://www.siam.org/journals/siims/8-2/99491.html>

[†]Department of Mathematics and Operational Research, Faculté Polytechnique, Université de Mons, 7000 Mons, Belgium (nicolas.gillis@umons.ac.be).

[‡]Department of Electronic Engineering, The Chinese University of Hong Kong, Sha Tin, N.T., Hong Kong (wkma@ee.cuhk.edu.hk).

where $X(:, j)$ is the spectral signature of the j th pixel, $W(:, k)$ is the spectral signature of the k th endmember, and $H(k, j)$ is the abundance of the k th endmember in the j th pixel so that $H \geq 0$ and $\|H(:, j)\|_1 = \sum_{i=1}^r |H(i, j)| = 1$ for all j (abundance sum-to-one constraint). Note that blind HU is closely related to NMF [32, 25], which aims at finding the best possible factorization of a nonnegative matrix $X \approx WH$, where W and H are nonnegative matrices. The only difference is the abundance sum-to-one constraint.

In blind HU, the so-called pure-pixel assumption plays a significant role. It is defined as follows. If for each endmember there exists a pixel containing only that endmember, that is, if for all $1 \leq k \leq r$ there exists j such that $X(:, j) = W(:, k)$, then the pure-pixel assumption holds. In that case, the matrix X has the following form:

$$X = W[L_r, H']\Pi, \quad \text{with } H' \geq 0, \|H'(:, j)\|_1 = 1 \quad \forall j,$$

and Π being a permutation. This implies that the columns of X are convex combinations of the columns of W , and hence blind HU under the linear mixing model and the pure-pixel assumption reduces to identifying the vertices of the convex hull of the columns of X ; see, e.g., [7, 14, 27] and the references therein. This problem is known to be efficiently solvable [4]. In the presence of noise, the problem becomes more difficult, and several provably robust algorithms have been proposed recently, for example the successive projection algorithm to be described in section 1.1. Note that, in the NMF literature, the pure-pixel assumption is referred to as the separability assumption [4], which has also been used successfully for text mining; see, e.g., [24, 3]. NMF under the separability assumption in the presence of noise is referred to as near-separable NMF; see, e.g., [16] and the references therein. Near-separable NMF is closely related to the more general archetypal analysis [13] that looks for basis elements (that is, columns of W) that are mixtures of data points; see also [11] and the references therein for recent developments.

Therefore, in this paper, we will assume that the matrix corresponding to the hyperspectral image has the following form (in the noiseless case).

Assumption 1 (separable matrix). The matrix X is separable if $X = WH \in \mathbb{R}^{m \times n}$, where $W \in \mathbb{R}^{m \times r}$ and $H = [L_r, H']\Pi \in \mathbb{R}_+^{r \times n}$, with the sum of the entries of each column of H' being at most one, that is, $\|H(:, j)\|_1 \leq 1$ for all j , and Π being a permutation.

Note that we have relaxed the assumption $\|H(:, j)\|_1 = 1$ for all j to $\|H(:, j)\|_1 \leq 1$ for all j ; this allows, for example, different illumination conditions among the pixels in the image. Note also that, in the context of separable NMF, the sum-to-one constraint $\|H(:, j)\|_1 = 1$ for all j can be made without loss of generality by normalizing each column of the input matrix M so that their entries sum to one; see, e.g., [10, 4].

1.1. Successive projection algorithm. The successive projection algorithm (SPA) is a simple but fast and robust pure-pixel search algorithm; see Algorithm SPA.

At each step of the algorithm, the column of the input matrix \tilde{X} with maximum ℓ_2 norm is selected, and then \tilde{X} is updated by projecting each column onto the orthogonal complement of the columns selected so far. SPA is extremely fast, as it can be implemented in $2mnr + \mathcal{O}(mr^2)$ operations [18]. SPA was first introduced in [2] and is closely related to other algorithms, such as automatic target generation process (ATGP), successive simplex volume maximization (SVMAX), and vertex component analysis (VCA); see the discussion

Algorithm SPA – Successive Projection Algorithm [2]

Input: Matrix $\tilde{X} = X + N$ with X satisfying Assumption 1, rank r .

Output: Set of r indices \mathcal{K} such that $\tilde{X}(:, \mathcal{K}) \approx W$.

- 1: Let $R = \tilde{X}$, $\mathcal{K} = \{\}$, $k = 1$.
 - 2: **while** $k \leq r$ and $R \neq 0$ **do**
 - 3: $p = \operatorname{argmax}_j \|R_{:,j}\|_2$.
 - 4: $R = \left(I - \frac{R_{:,p}R_{:,p}^T}{\|R_{:,p}\|_2^2} \right) R$.
 - 5: $\mathcal{K} = \mathcal{K} \cup \{p\}$.
 - 6: $k = k + 1$.
 - 7: **end while**
-

in [27]. What makes SPA distinguishingly interesting is that it is provably robust against noise [18].

Theorem 1.1 (see [18, Thm. 3]). *Let $\tilde{X} = X + N$, where X satisfies Assumption 1, W has full column rank, and N is noise with $\max_j \|N(:, j)\|_2 \leq \epsilon$. If $\epsilon \leq \mathcal{O}\left(\frac{\sigma_{\min}(W)}{\sqrt{r}\kappa^2(W)}\right)$, then SPA identifies the columns of W up to error $\mathcal{O}(\epsilon\kappa^2(W))$; that is, the index set \mathcal{K} identified by SPA satisfies*

$$\max_{1 \leq j \leq r} \min_{k \in \mathcal{K}} \left\| W(:, j) - \tilde{X}(:, k) \right\|_2 \leq \mathcal{O}(\epsilon\kappa^2(W)),$$

where $\kappa(W) = \frac{\sigma_{\max}(W)}{\sigma_{\min}(W)}$ is the condition number of W , and $\|x\|_2 = \sqrt{\sum_{i=1}^n x_i^2}$ for $x \in \mathbb{R}^n$.

1.2. Preconditioning. If a matrix X satisfying Assumption 1 is premultiplied by a matrix Q , it still satisfies Assumption 1, where W is replaced with QW . Since pure-pixel search algorithms are sensitive to the conditioning of matrix W , it would be beneficial to find a matrix Q that reduces the conditioning of W . In particular, the robustness result of SPA (Theorem 1.1) can be adapted when the input matrix is premultiplied by a matrix Q :

Corollary 1.2. *Let $\tilde{X} = X + N$, where X satisfies Assumption 1, W has full column rank, and N is noise with $\max_j \|N(:, j)\|_2 \leq \epsilon$; and let $Q \in \mathbb{R}^{p \times m}$ ($p \geq r$). If QW has full column rank and*

$$\epsilon \leq \mathcal{O}\left(\frac{\sigma_{\min}(W)}{\sqrt{r}\kappa^3(QW)}\right),$$

then SPA applied on matrix $Q\tilde{X}$ identifies indices corresponding to the columns of W up to error $\mathcal{O}(\epsilon\kappa(W)\kappa(QW)^3)$.

Proof. This result follows directly from [19, Cor. 1]. In fact, in [19, Cor. 1], the result is proved for

$$\epsilon \leq \mathcal{O}\left(\frac{\sigma_{\min}(QW)}{\sqrt{r}\sigma_{\max}(Q)\kappa^2(QW)}\right)$$

with error up to $\mathcal{O}(\epsilon\kappa(Q)\kappa(QW)^2)$. Since

$$\frac{\sigma_{\min}(QW)}{\sigma_{\max}(Q)} \geq \frac{\sigma_{\min}(QW)}{\sigma_{\max}(QW)\sigma_{\max}(W^{-1})} = \frac{\sigma_{\min}(W)}{\kappa(QW)}$$

and

$$\kappa(Q) = \kappa(QWW^{-1}) \leq \kappa(QW)\kappa(W^{-1}) = \kappa(QW)\kappa(W),$$

the proof is complete. \blacksquare

Note that Corollary 1.2 does not simply amount to replacing W by QW in Theorem 1.1 because the noise N is also premultiplied by Q . Note also that, in view of Theorem 1.1, preconditioning is beneficial for any Q such that $\kappa(QW)^3 \leq \kappa(W)$.

1.2.1. Semidefinite program–based preconditioning. Assume that $m = r$ (the problem can be reduced to this case using noise filtering; see section 3) and that W is full rank. An optimal preconditioning would be $Q = W^{-1}$ so that $QW = I_r$ would be perfectly conditioned, that is, $\kappa(QW) = 1$. In particular, applying Corollary 1.2 with $Q = W^{-1}$ gives the following result: if $\epsilon \leq \mathcal{O}(\frac{\sigma_{\min}(W)}{\sqrt{r}})$, then SPA applied on matrix $Q\tilde{X}$ identifies indices corresponding to the columns of W up to error $\mathcal{O}(\epsilon\kappa(W))$. This is a significant improvement compared to Theorem 1.1, especially for the upper bound on the noise level: the term $\kappa(W)^2$ disappears from the denominator. For hyperspectral images, $\kappa(W)$ can be rather large, as spectral signatures often share similar patterns. Note that the bound on the noise level is essentially optimal for SPA since $\epsilon \geq \Omega(\sigma_{\min}(W))$ would allow the noise to make the matrix W rank deficient [19].

Of course, W^{-1} is unknown; otherwise the problem would be solved. However, it turns out that it is possible to compute W^{-1} approximately (up to orthogonal transformations, which do not influence the conditioning) even in the presence of noise using the minimum volume ellipsoid centered at the origin containing all columns of \tilde{X} [19]. An ellipsoid \mathcal{E} centered at the origin in \mathbb{R}^r is described via a positive definite matrix $A \in \mathbb{S}_{++}^r$: $\mathcal{E} = \{x \in \mathbb{R}^r | x^T A x \leq 1\}$. The volume of \mathcal{E} is equal to $\det(A)^{-1/2}$ times the volume of the unit ball in dimension r . Therefore, given a matrix $\tilde{X} \in \mathbb{R}^{r \times n}$ of rank r , we can formulate the minimum volume ellipsoid centered at the origin and containing the columns \tilde{x}_j for all j of matrix \tilde{X} as follows:

$$(1.1) \quad A^* = \underset{A \in \mathbb{S}_{++}^r}{\operatorname{argmax}} \det(A) \text{ such that } \tilde{x}_j^T A \tilde{x}_j \leq 1 \quad \forall j.$$

This problem can be formulated as a semidefinite program (SDP) [8, p. 222]. It was shown in [19] that

- in the noiseless case (that is, $N = 0$), the optimal solution A^* of (1.1) is given by $(WW^T)^{-1}$, and hence factoring A^* allows one to recover W^{-1} (up to orthogonal transformations), and that
- in the noisy case, the optimal solution of (1.1) is close to $(WW^T)^{-1}$ and hence leads to a good preconditioning for SPA.

More precisely, the following robustness result was proved.

Theorem 1.3 (see [19, Thm. 3]). *Let $\tilde{X} = X + N$, where X satisfies Assumption 1 with $m = r$, W has full column rank, and N is noise with $\max_j \|N(:, j)\|_2 \leq \epsilon$. If $\epsilon \leq \mathcal{O}(\frac{\sigma_{\min}(W)}{r\sqrt{r}})$, then SDP-based preconditioned SPA identifies a subset \mathcal{K} so that $\tilde{X}(:, \mathcal{K})$ approximates the columns of W up to error $\mathcal{O}(\epsilon\kappa(W))$.*

In case $m > r$, it was proposed to first replace the data points by their projections onto the r -dimensional linear subspace obtained with the SVD (that is, use a linear dimensionality reduction technique for noise filtering; see also section 3); see Algorithm SDP-Prec.

Algorithm SDP-Prec – SDP-based Preconditioning [19]

Input: Matrix $\tilde{X} = X + N$ with X satisfying Assumption 1, rank r .

Output: Preconditioner Q .

- 1: $[U_r, \Sigma_r, V_r] = \text{rank-}r \text{ truncated SVD}(\tilde{X})$.
 - 2: Let $\tilde{X} \leftarrow \Sigma_r V_r^T$ and solve (1.1) to get A^* .
 - 3: Factorize $A^* = P^T P$ (e.g., Cholesky decomposition).
 - 4: $Q = P U_r^T$.
-

Algorithm **SDP-Prec** first requires the truncated SVD, which can be computed in $\mathcal{O}(mnr)$ operations. It then requires the solution of a particular SDP ($\mathcal{O}(r^2)$ variables and n constraints—note that $n \geq r$; otherwise the optimal value of the problem is unbounded) that can be solved in $\mathcal{O}(n^{3.5} \log(n/\epsilon))$ operations to obtain an ϵ -solution [23] (ϵ is a relative measure of nonoptimality); see also [34] and the discussion therein for more details. However, in practice, effective active set methods can be used to solve large-scale problems [34]: in fact, one can keep only $\frac{r(r+1)}{2}$ constraints from (1.1) to obtain an equivalent problem [22].

Note that Mizutani [28] solves the same SDP, but for another purpose, namely to preprocess the input matrix by removing the columns which are not on the boundary of the minimum volume ellipsoid.

1.3. Motivation and contribution of the paper. Although the pure-pixel assumption is rather strong and not perfectly satisfied in many scenarios (because of low resolution, mixed materials, variability across the image, nonlinearities, etc.), pure-pixel search algorithms are still extremely useful and widely used in practice because they are fast, simple to use, and are able to recover endmembers in many situations. They can also be used, for example, as initialization strategies for more sophisticated techniques. Hence it is important to gain theoretical insights on these techniques to improve their understanding and work towards more effective variants. The focus of this paper is on a *theoretical analysis of the robustness to noise of several preconditionings for pure-pixel search algorithms* or, equivalently, for near-separable NMF algorithms.

The SDP-based preconditioning described in section 1.2 is appealing in the sense that it builds an approximation of W^{-1} whose error is provably bounded by the noise level. This is a somewhat ideal solution but can be computationally expensive to obtain since it requires the resolution of an SDP. Hence, a natural move is to consider computationally cheaper preconditioning alternatives. The contribution of this paper is threefold:

1. In section 2, we analyze the robustness of preconditionings obtained using approximate solutions of (1.1) and prove the following (see Theorem 2.6 for all the details): under the same assumptions as in Theorem 1.3 but using a preconditioning obtained with a feasible point of (1.1) whose objective function value is some multiplicative factor α away from the optimal value (which we refer to as an α -approximate solution) and if $\epsilon \leq \mathcal{O}(\min(\frac{1}{r}, \alpha^{3/2}) \frac{\sigma_{\min}(W)}{\sqrt{r}})$, then SPA identifies indices corresponding to the columns of W up to error $\mathcal{O}(\epsilon \kappa(W) \alpha^{-3/2})$. This shows that any good feasible point of (1.1) provides a reasonable preconditioning, which (i) gives a theoretical motivation

for developing less accurate but faster solvers for (1.1), and (ii) will be useful for analyzing two other preconditionings described in sections 3 and 4.

2. In section 3, we analyze the robustness of prewhitening, a standard preconditioning technique in blind source separation. We try to understand under which conditions prewhitening can be as good as the SDP-based preconditioning; in fact, prewhitening was shown to perform very similarly to the SDP-based preconditioning on some synthetic data sets [19]. Prewhitening corresponds to a solution of (1.1) with additional constraints, and hence the robustness of prewhitened SPA follows from the result above (section 3.2). However, this result is not tight, and we provide a tight robustness analysis of prewhitening (section 3.3). We also provide a robustness analysis of prewhitening under a standard generative model (section 3.4).
3. In section 4, we analyze a preconditioning based on SPA itself. The idea was proposed in [17], where the resulting method was found to be extremely fast and, as opposed to prewhitening, to perform perfectly in the noiseless case and not be affected by the abundances of the different endmembers. Moreover, we are able to improve the theoretical bound on the noise level allowed by SPA by a factor $\kappa(W)$ using this preconditioning.

Finally, in section 5, we summarize our theoretical findings in Table 1 and illustrate these results on synthetic data sets. In particular, we show that prewhitening and the SPA-based preconditioning perform competitively with the SDP-based preconditioning while showing much better runtime performance in practice.

2. Analysis of approximate SDP preconditioning. In this section, we analyze the effect of using an approximate solution A of (1.1) instead of the optimal one A^* for preconditioning matrix \tilde{X} . We will say that A is an α -approximate solution of (1.1) for some $0 < \alpha \leq 1$ if A is a feasible point of (1.1), that is, $A \in \mathbb{S}_+^r$ and $\tilde{x}_j^T A \tilde{x}_j \leq 1$ for all j , and

$$\det(A) \geq \alpha \det(A^*),$$

where A^* is the optimal solution of (1.1). Letting $A = QQ^T$, analyzing the effect of Q as a preconditioning reduces to showing that QW is well-conditioned, that is, to upper bound $\kappa(QW)$, which is equivalent to bounding $\kappa(W^T AW)$ since $\kappa(W^T AW) = \kappa(QW)^2$. In fact, if $\kappa(QW)$ can be bounded, the robustness of preconditioned SPA follows from Corollary 1.2.

In [19], a change of variables is performed on the SDP (1.1) using $A = W^{-T}CW^{-1}$ to obtain the following equivalent problem:

$$(2.1) \quad C^* = \operatorname{argmax}_{C \in \mathbb{S}_+^r} \det(C) \det(W)^{-2} \quad \text{such that} \quad \tilde{x}_j^T (W^{-T}CW^{-1}) \tilde{x}_j \leq 1 \quad \forall j.$$

Since our goal is to bound $\kappa(W^T AW)$ and $W^T AW = C$, it is equivalent to bound $\kappa(C)$. It was shown in [19] that, for sufficiently small noise level ϵ , $\kappa(W^T A^* W) = \kappa(C^*) = \mathcal{O}(1)$, which implies the robustness of SDP-preconditioned SPA; see Theorem 1.3. In this section, we analyze (2.1) directly and will use the following assumption.

Definition 2.1. *The matrix C is an α -approximate point of (2.1) for some $0 < \alpha \leq 1$; that is, C is a feasible point of (2.1) and*

$$\det(C) \geq \alpha \det(C^*).$$

Note that C is an α -approximate solution of (2.1) if and only if $A = W^{-T}CW^{-1}$ is an α -approximate solution of (1.1). In fact, $\det(W^{-T}CW^{-1}) = \det(W^{-T})\det(C)\det(W^{-1})$.

The main result of this section is therefore to show that any α -approximate solution of (2.1) is well-conditioned. More precisely, we show in Theorem 2.5 that, for a sufficiently small noise level ϵ ,

$$\kappa(C) \leq \frac{12}{\alpha}.$$

This will imply, by Corollary 1.2, that preconditioned SPA using an approximate solution of (1.1) is robust to noise, given that α is sufficiently close to one; see Theorem 2.6.

2.1. Bounding $\kappa(C)$. It is interesting to notice that the case $r = 1$ is trivial since C^* is a scalar and thus has $\kappa(C) = 1$ (in fact, all columns of \tilde{X} are multiples of the unique column of $W \in \mathbb{R}^{m \times 1}$). Otherwise, since $\kappa(C) \geq 1$, we need only to provide an upper bound for $\kappa(C)$. The steps of the proof are the following:

- Derive a lower bound for $\det(C)$ (Lemma 2.2).
- Provide an upper bound of $\text{tr}(C)$ (Lemma 2.3).
- Combine the two bounds above to bound $\kappa(C)$. In fact, we prove in Lemma 2.4 that the condition number $\kappa(C)$ of an r -by- r matrix C with $\text{tr}(C) \leq \beta$ and $\det(C) \geq \gamma$ can be bounded above; see (2.3).

The lower bound for $\det(C)$ and the upper bound for $\text{tr}(C)$ follow directly from results in [19].

Lemma 2.2. *If $\tilde{X} = W + N$, where X satisfies Assumption 1 with $m = r$, then any α -approximate solution C of (2.1) satisfies*

$$(2.2) \quad \det(C) \geq \alpha \left(1 + \frac{\epsilon}{\sigma_{\min}(W)} \right)^{-2r}.$$

Proof. In [19, Lemma 1], it was proved that the optimal solution C^* of (2.1) satisfies

$$\det(C^*) \geq \left(1 + \frac{\epsilon}{\sigma_{\min}(W)} \right)^{-2r}.$$

Hence, the result follows directly from Definition 2.1. ■

Lemma 2.3. *If $\tilde{X} = W + N$, where X satisfies Assumption 1 with $m = r$, and*

$$\epsilon \leq \frac{\sigma_{\min}(W)}{8r\sqrt{r}},$$

then any feasible point C of (2.1) satisfies $\text{tr}(C) \leq r + 1$.

Proof. See Lemma 2 and the proof of Lemma 3 in [18]. ■

Lemma 2.4. *The optimal value*

$$\kappa^* = \max_{\lambda \in \mathbb{R}^r} \frac{\lambda_1}{\lambda_r} \text{ such that } \sum_{i=1}^r \lambda_i \leq \beta, \prod_{i=1}^r \lambda_i \geq \gamma, \text{ and } \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_r \geq 0,$$

where $\beta \geq r$ and $0 < \gamma \leq 1$ is given by

$$(2.3) \quad \kappa^* = \frac{1 + \sqrt{1 - \gamma \left(\frac{r}{\beta} \right)^r}}{1 - \sqrt{1 - \gamma \left(\frac{r}{\beta} \right)^r}}.$$

Proof. The proof is given in Appendix A. ■

Theorem 2.5. *If $\tilde{X} = X + N$, where X satisfies Assumption 1 with $m = r \geq 2$ and $\max_j \|N(:, j)\|_2 \leq \epsilon \leq \frac{\sigma_{\min}(W)}{8r\sqrt{r}}$, then*

$$\kappa(C) \leq \frac{12}{\alpha},$$

where C is an α -approximate solution of (2.1).

Proof. By Lemma 2.2, we have

$$\det(C) \geq \alpha \left(1 + \frac{\epsilon}{\sigma_{\min}(W)}\right)^{-2r} \geq \alpha \left(1 + \frac{1}{8r\sqrt{r}}\right)^{-2r} \geq \alpha \underbrace{\left(1 + \frac{1}{64}\right)^{-4}}_{=\eta} = \alpha\eta,$$

where the third inequality above is obtained by the fact that $\left(1 + \frac{1}{8r\sqrt{r}}\right)^{-2r}$ is increasing in r for $r \geq 1$. Also, by Lemma 2.3, we have $\text{tr}(C) \leq r + 1$. Combining these results with Lemma 2.4 (via setting $\gamma = \eta\alpha$ and $\beta = r + 1$) yields

$$\kappa(C) \leq \frac{1 + \sqrt{1 - \alpha\eta \left(\frac{r}{r+1}\right)^r}}{1 - \sqrt{1 - \alpha\eta \left(\frac{r}{r+1}\right)^r}} \leq \frac{1 + \sqrt{1 - \frac{\alpha}{3}}}{1 - \sqrt{1 - \frac{\alpha}{3}}},$$

where the second inequality above is obtained by the fact that $\left(\frac{r}{r+1}\right)^r$ is nonincreasing in r and its limit is given by

$$\lim_{r \rightarrow \infty} \left(\frac{r}{r+1}\right)^r = \lim_{r \rightarrow \infty} \left(1 - \frac{1}{r+1}\right)^r = \frac{1}{e} = 0.3679$$

and that $\frac{\eta}{e} = \frac{(1+\frac{1}{64})^{-4}}{e} \geq \frac{1}{3}$. Finally, since the function $\frac{1+y}{1-y}$ is increasing for $0 \leq y < 1$ and $\sqrt{1-x} \leq 1 - \frac{x}{2}$ for all $0 \leq x \leq 1$, we have

$$\frac{1 + \sqrt{1 - \frac{\alpha}{3}}}{1 - \sqrt{1 - \frac{\alpha}{3}}} \leq \frac{1 + 1 - \frac{\alpha}{6}}{1 - 1 + \frac{\alpha}{6}} \leq \frac{12}{\alpha} - 1. \quad \blacksquare$$

2.2. Robustness of SPA preconditioned with an approximate SDP solution. The upper bound on $\kappa(C)$ (Theorem 2.5) proves that the preconditioning generates a well-conditioned near-separable matrix for α sufficiently close to one. Hence any good feasible point of (1.1) allows us to obtain more robust near-separable NMF algorithms. In particular, we have the following result for SPA.

Theorem 2.6. *Let $\tilde{X} = X + N$, where X satisfies Assumption 1 with $m = r$, W has full rank, and the noise N satisfies $\max_j \|N(:, j)\|_2 \leq \epsilon$. Let also $Q \in \mathbb{R}^{r \times r}$ be such that $A = Q^T Q$, where A is an α -approximate solution of (1.1). If*

$$\epsilon \leq \mathcal{O} \left(\min \left(\frac{1}{r}, \alpha^{3/2} \right) \frac{\sigma_{\min}(W)}{\sqrt{r}} \right),$$

then SPA applied on matrix $Q\tilde{X}$ identifies indices corresponding to the columns of W up to error $\mathcal{O}(\epsilon\kappa(W)\alpha^{-3/2})$.

Proof. This follows from Corollary 1.2 and Theorem 2.5. Let Q be such that $A = Q^T Q$, where A is an α -approximate solution of (1.1). Since $C = W^T A W$ is an α -approximate solution of (2.1), we have that

$$(QW)^T QW = W^T Q^T QW = W^T A W = C.$$

Using $\sigma_i(QW) = \sqrt{\lambda_i(C)}$ for all i and by Theorem 2.5, for which we need to assume that $\epsilon \leq \mathcal{O}(\frac{\sigma_{\min}(W)}{r\sqrt{r}})$, we further obtain

$$\kappa(QW) \leq \sqrt{12\alpha^{-1}} = 2\sqrt{3}\alpha^{-1/2}.$$

Applying the above bound to Corollary 1.2 leads to the desired result: $\epsilon \leq \mathcal{O}(\frac{\sigma_{\min}(W)}{\kappa^3(QW)\sqrt{r}})$ leads to an error proportional to $\mathcal{O}(\epsilon\kappa(W)\kappa^3(QW))$. ■

Theorem 2.6 shows that the SDP-based preconditioning does not require a high accuracy solution of the SDP. For example, compared to the optimal solution, any $\frac{1}{2}$ -approximate solution would not change the upper bound on the noise level for $r \geq 3$ (since $0.5^{3/2} \geq 1/3$) while it would increase the error by a factor smaller than three. This gives a theoretical motivation for developing less accurate but faster solvers for (1.1). For example, one could use the first-order proximal point algorithm proposed in [35] or the modified Frank–Wolfe algorithm in [1], with cheaper per-iteration cost (but slower convergence than interior point methods). One could also use active-set methods with an early termination criterion: usually, keeping a well-chosen subset of the data points already leads to a good feasible point of (1.1), up to a scaling factor. In fact, as mentioned in the introduction, only $\frac{r(r+1)}{2}$ constraints are actually necessary [22]. Moreover, keeping only dr constraints already allows us to obtain an approximate solution with optimality guarantees that can be computed in $\mathcal{O}(dr^3n)$ [5, Thm. 3.1]: for example, keeping $4r$ constraints leads to a $1/9^r$ -approximate solution that can be computed in $\mathcal{O}(r^3n)$ operations; see also the discussion in [29]. Developing and comparing the corresponding preconditionings is out of the scope of this paper and is a direction for further research.

Also, this analysis will allow us to better understand two other preconditionings: prewhitening (section 3) and SPA-based preconditioning (section 4).

3. Prewhitening. Noise filtering and prewhitening are standard techniques in blind source separation; see, e.g., [12]. They were used in [19] as a preconditioning for SPA. In this section, in light of the results from the previous section, we analyze prewhitening as a preconditioning. We bound the condition number of $\kappa(QW)$, where Q is the preconditioner obtained from prewhitening. In the worst case, $\kappa(QW)$ can be large, as it depends on the number of pixels n . However, under some standard generative model, prewhitening can be shown to be much more robust.

3.1. Description. Let $(U_r, \Sigma_r, V_r) \in \mathbb{R}^{m \times r} \times \mathbb{R}^{r \times r} \times \mathbb{R}^{n \times r}$ be the rank- r truncated SVD of $\tilde{X} = X + N$ so that $\tilde{X}_r = U_r \Sigma_r V_r^T$ is the best rank- r approximation of \tilde{X} with respect to the Frobenius norm. Assuming that the noiseless data matrix X lives in an r -dimensional

linear space and assuming Gaussian noise, replacing \tilde{X} with \tilde{X}_r allows noise filtering. Given \tilde{X}_r , prewhitening amounts to keeping only the matrix V_r . Equivalently, it amounts to pre-multiplying \tilde{X}_r (or \tilde{X}) with $Q = \Sigma_r^{-1}U_r^T$ since

$$Q\tilde{X} = (\Sigma_r^{-1}U_r^T)(U\Sigma V^T) = (\Sigma_r^{-1}U_r^T)(U_r\Sigma_r V_r^T) = Q\tilde{X}_r = V_r^T,$$

where (U, Σ, V) is the full SVD of \tilde{X} (recall that $U(:, 1:r) = U_r$ and that the columns of U and V are orthonormal); see Algorithm NF-PW.

Algorithm NF-PW – Noise Filtering and Prewhitening

Input: Matrix $\tilde{X} = X + N$ with X satisfying Assumption 1, rank r .

Output: Preconditioner Q .

- 1: $[U_r, \Sigma_r, V_r] = \text{rank-}r \text{ truncated SVD}(\tilde{X})$.
 - 2: $Q = \Sigma_r^{-1}U_r^T$.
-

In [19], Algorithm NF-PW is used as an heuristic preconditioning and performs similarly to the SDP-based preconditioning. Algorithm NF-PW requires $\mathcal{O}(mnr)$ operations to compute the rank- r truncated SVD of \tilde{X} .

3.2. Link with the SDP-based preconditioning. For simplicity, let us consider the case $m = r$ (or, equivalently, assume that noise filtering has already been applied to the data matrix). In that case, the preconditioner Q given by prewhitening is

$$Q = \Sigma_r^{-1}U_r^T = \left(\tilde{X}\tilde{X}^T\right)^{-1/2},$$

where $(\cdot)^{-1/2}$ denotes the inverse of the square root of a positive definite matrix, that is, $Q = S^{-1/2} \iff S = (Q^T Q)^{-1}$ (which is unique up to orthogonal transformations). In fact, for $m = r$, $\tilde{X} = U_r \Sigma_r V_r^T$, and hence $\tilde{X}\tilde{X}^T = (U_r \Sigma_r)(U_r \Sigma_r)^T$. We have the following well-known result.

Lemma 3.1. *Let $\tilde{X} \in \mathbb{R}^{r \times n}$ be of rank r . Then $B^* = (\tilde{X}\tilde{X}^T)^{-1}$ is the optimal solution of*

$$(3.1) \quad \max_{B \in \mathbb{S}_+^r} \det(B) \quad \text{such that} \quad \sum_{i=1}^n \tilde{x}_i^T B \tilde{x}_i \leq r.$$

Moreover, B^* satisfies $\tilde{x}_j^T B^* \tilde{x}_j \leq 1$ for all j .

Proof. Observing that the objective can be replaced with $\log \det(B)$ and that the constraint will be active at optimality (otherwise B can be multiplied by a scalar larger than one), any optimal solution has to satisfy the following first-order optimality conditions:

$$B^{-1} = \lambda \sum_{i=1}^n \tilde{x}_i \tilde{x}_i^T = \lambda \tilde{X}\tilde{X}^T \quad \text{and} \quad \sum_{i=1}^n \tilde{x}_i^T B \tilde{x}_i = r,$$

where $\lambda \geq 0$ is the Lagrangian multiplier. This implies that $B^* = \lambda^{-1}(\tilde{X}\tilde{X}^T)^{-1}$. We have

$$\sum_{i=1}^n \tilde{x}_i^T B^* \tilde{x}_i = \lambda^{-1} \left\langle \left(\tilde{X}\tilde{X}^T\right)^{-1}, \tilde{X}\tilde{X}^T \right\rangle = \lambda^{-1} \text{tr} \left(\left(\tilde{X}\tilde{X}^T\right)^{-1} \tilde{X}\tilde{X}^T \right) = \lambda^{-1} \text{tr}(I_r) = r\lambda^{-1}.$$

The above equation, together with the condition $\sum_{i=1}^n \tilde{x}_j^T B \tilde{x}_j = r$, implies that $\lambda = 1$, which proves $B^* = (\tilde{X} \tilde{X}^T)^{-1}$.

Denoting $\tilde{X} = U \Sigma V^T$ the compact SVD of \tilde{X} ($V \in \mathbb{R}^{n \times r}$), we obtain $\tilde{x}_j^T B \tilde{x}_j = \|V(j, :)\|_2^2$, which is smaller than one since V has orthogonal columns and $n \geq r$. ■

A robustness analysis of prewhitening follows directly from Theorem 2.6. In fact, by Lemma 3.1, the matrix $(\tilde{X} \tilde{X}^T)^{-1}$ is a feasible point of the minimum volume ellipsoid problem (1.1). Moreover, it is optimal up to a factor $\frac{n}{r}$: by Lemma 3.1, the optimal solution of

$$\max_{D \in \mathbb{S}_+^r} \det(D) \quad \text{such that} \quad \sum_{i=1}^n \tilde{x}_i^T D \tilde{x}_i \leq n$$

is given by $D^* = \frac{n}{r} B^*$ while the optimal solution A^* of (1.1) is a feasible point of this problem (since $\tilde{x}_j^T A^* \tilde{x}_j \leq 1$ for all j) so that $\det(B^*) \leq \det(A^*)$ and

$$\det(A^*) \leq \det(D^*) = \det\left(\frac{n}{r} B^*\right) = \left(\frac{n}{r}\right)^r \det(B^*),$$

and hence

$$\det(B^*) \geq \left(\frac{r}{n}\right)^r \det(A^*).$$

In other words, B^* is an $\left(\frac{r}{n}\right)^r$ -approximate solution of (1.1). Combining this result with Theorem 2.6, we obtain the following corollary.

Corollary 3.2. *Let $\tilde{X} = X + N$, where X satisfies Assumption 1 with $m = r$, W is of full rank, and the noise N satisfies $\max_j \|N(:, j)\|_2 \leq \epsilon$. If*

$$\epsilon \leq \mathcal{O}\left(\left(\frac{r}{n}\right)^{\frac{3r}{2}} \frac{\sigma_{\min}(W)}{\sqrt{r}}\right),$$

prewhitened SPA identifies up to error $\mathcal{O}(\epsilon \kappa(W) \left(\frac{n}{r}\right)^{\frac{3r}{2}})$ r indices corresponding to the columns of W .

This bound is rather bad, as n is often large compared to r ; for the HU application, we typically have $n \geq 10^6$ and $r \lesssim 30$. In the next subsections, we provide a tight robustness analysis of prewhitening and analyze prewhitening under a standard generative model.

3.3. Tight robustness analysis. In this subsection, we provide a better robustness analysis of prewhitening. More precisely, we provide a tight upper bound for $\kappa(QW)$. As before, we consider only the case $m = r$. Under Assumption 1, we have

$$\tilde{X} = X + N = W[I_r, H'] + N = W([I_r, H'] + W^{-1}N) = W([I_r, H'] + N') = WY,$$

where we denote $N' = W^{-1}N$ and $Y = [I_r, H'] + N'$. Recall that the conditioner Q given by prewhitening is $Q = (\tilde{X} \tilde{X}^T)^{-1/2}$. Hence, the condition number of QW will be equal to the square root of the condition number of Y . In fact,

$$Q = \left(\tilde{X} \tilde{X}^T\right)^{-1/2} = (WY Y^T W^T)^{-1/2} = (Y Y^T)^{-1/2} W^{-1}$$

so that $\kappa(QW) = \kappa((YY^T)^{-1/2}) = \kappa(Y)$. Therefore, to provide a robustness analysis of prewhitening, it is sufficient to bound $\kappa(Y)$. In the next lemma, we show that $\kappa(Y) \leq \mathcal{O}(\sqrt{n-r+1})$, which implies that $\kappa(QW) \leq \mathcal{O}(\sqrt{n-r+1})$ and will lead to an error bound that is much smaller than that derived in the previous subsection.

Lemma 3.3. *Let $H' \in \mathbb{R}^{r \times (n-r)}$ be a nonnegative matrix with $\|H'(:,j)\|_1 \leq 1$ for all j , and let $N' \in \mathbb{R}^{r \times n}$ satisfy $\|N'\|_2 \leq \delta < 1$. Then,*

$$\kappa([I_r, H'] + N') \leq \frac{\sqrt{1+n-r} + \delta}{1-\delta}.$$

Proof. Let us first show that

$$\sigma_{\min}([I_r, H']) \geq 1 \quad \text{and} \quad \sigma_{\max}([I_r, H']) \leq \sqrt{1+n-r}.$$

We have

$$\sigma_p([I_r, H'])^2 = \lambda_p([I_r, H'][I_r, H']^T),$$

where $p = \min$ or \max , and note that

$$[I_r, H'][I_r, H']^T = I_r + H'H'^T.$$

Using $H'H'^T \succeq 0$, one easily gets

$$\lambda_{\min}(I_r + H'H'^T) \geq \lambda_{\min}(I_r) = 1.$$

Also, we have

$$\lambda_{\max}(I_r + H'H'^T) \leq \lambda_{\max}(I_r) + \lambda_{\max}(H'H'^T) \leq 1 + n - r,$$

as $\lambda_{\max}(H'H'^T) \leq \text{tr}(H'H'^T)$ and

$$\text{tr}(H'H'^T) = \text{tr}\left(\sum_j H'(:,j)H'(:,j)^T\right) = \sum_j \text{tr}\left(H'(:,j)H'(:,j)^T\right) = \sum_j \|H'(:,j)\|_2^2 \leq n - r,$$

since $\|H'(:,j)\|_2 \leq \|H'(:,j)\|_1 \leq 1$. Finally, using the singular value perturbation theorem (Weyl; see, e.g., [20]), we have

$$\sigma_p([I_r, H'] - \|N\|_2) \leq \sigma_p([I_r, H'] + N) \leq \sigma_p([I_r, H']) + \|N\|_2$$

for $p = \min$ or \max , and since $\|N\|_2 \leq \delta < 1$, we obtain

$$\kappa([I_r, H'] + N) \leq \frac{\sqrt{1+n-r} + \delta}{1-\delta}. \quad \blacksquare$$

This bound allows us to provide a robustness analysis of prewhitening.

Theorem 3.4. *Let $\tilde{X} = X + N$, where X satisfies Assumption 1 with $m = r$, W has full rank, and the noise N satisfies $\max_j \|N(:,j)\|_2 \leq \epsilon$. If $\epsilon < \mathcal{O}\left(\frac{\sigma_{\min}(W)}{(n-r+1)^{3/2}\sqrt{r}}\right)$, prewhitened SPA identifies the columns of W up to error $\mathcal{O}\left((n-r+1)^{3/2}\epsilon\kappa(W)\right)$.*

Proof. This follows from Corollary 1.2 and Lemma 3.3. We have

$$\|N'\|_2 = \|W^{-1}N\|_2 \leq \frac{\|N\|_2}{\sigma_{\min}(W)} \leq \frac{\sqrt{n} \max_j \|N(:,j)\|_2}{\sigma_{\min}(W)} \leq \epsilon \frac{\sqrt{n}}{\sigma_{\min}(W)} \leq \mathcal{O}\left(\frac{1}{n}\right),$$

and as a result, Lemma 3.3 can be applied to obtain $\kappa(QW) \leq \mathcal{O}(\sqrt{n-r+1})$ for prewhitening. By plugging the above bound into Corollary 1.2, Theorem 3.4 is obtained. ■

The bounds of Theorem 3.4 are tight. In fact, if, except for the pure pixels, all pixels contain the same endmember, say the k th, then all columns of H' are equal to the k th column of the identity matrix, that is, $H'(:,j) = I_r(:,k) := e_k$ for all j . Therefore,

$$\kappa(Y) = \kappa([I_r, H']) = \sqrt{1+n-r}$$

since $YY^T = (I_r + H'H^T) = I_r + (n-r)e_k e_k^T$ is a diagonal matrix with $(YY^T)_{ii} = 1$ for all $i \neq k$ and $(YY^T)_{kk} = 1+n-r$.

This indicates that prewhitening should perform the worst when one endmember contains most pixels, as this matches the upper bound of Theorem 3.4. However, if the pixels are relatively well spread in the convex hull of the endmembers, then prewhitening may perform well. This will be proved in the next subsection, wherein the robustness of prewhitening under a standard generative model is analyzed.

3.4. Robustness under a standard generative model. We continue our analysis by considering a standard generative model in HU. We again consider $m = r$, and the generative model is described as follows.

Assumption 2. The near-separable matrix $\tilde{X} = WH + N$ is such that the following hold.

- (i) W is of full rank.
- (ii) $H(:,j)$ is independent and identically distributed (i.i.d.) following a Dirichlet distribution with parameter $\rho = (\rho_1, \dots, \rho_r) > 0$ for all j . Also, without loss of generality, it will be assumed that $\rho_1 \geq \rho_2 \geq \dots \geq \rho_r$.
- (iii) $N(:,j)$ is i.i.d. with mean zero and covariance $\mathbb{E}[N(:,j)N(:,j)^T] = \sigma_N^2 I$ for all j (Gaussian noise).
- (iv) The number of samples goes to infinity; that is, $n \rightarrow \infty$.

Note that assumption (ii), which models the abundances as being Dirichlet distributed, is a popular assumption in the HU context; see, e.g., [31]. In particular, the parameter ρ characterizes how the pixels are spread. To describe this, let us consider a simplified case where $\beta := \rho_1 = \dots = \rho_r$, i.e., symmetric Dirichlet distribution. We have the following phenomena: if $\beta = 1$, then $H(:,j)$'s are uniformly distributed over the unit simplex; if $\beta < 1$ and β decreases, then $H(:,j)$'s are more concentrated around the vertices (or pure pixels) of the simplex; if $\beta > 1$ and β increases, then $H(:,j)$'s are more concentrated around the center of the simplex. In fact, $\beta \rightarrow 0$ means that $H(:,j)$'s contain only pure pixels in the same proportions. It should also be noted that we do not assume the separability or pure-pixel assumption, although the latter is implicitly implied by Assumption 2. Specifically, under assumptions (ii) and (iv), for every endmember there exist pixels that are arbitrarily close to the pure pixel in a probability one sense.

Now, our task is to prove a bound on $\kappa(QW)$ under the above statistical assumptions, thereby obtaining implications on how prewhitened SPA may perform with respect to the

abundances' distribution (rather than in the worst-case scenario). To proceed, we formulate the prewhitening preconditioner as

$$Q = R^{-1/2}, \text{ where } R = \frac{1}{n} \tilde{X} \tilde{X}^T = \frac{1}{n} \sum_{j=1}^n \tilde{X}(:,j) \tilde{X}(:,j)^T.$$

For $n \rightarrow \infty$, we have

$$R = \mathbb{E} \left[\tilde{X}(:,j) \tilde{X}(:,j)^T \right].$$

Also, under Assumption 2, the above correlation matrix can be shown to be

$$\mathbb{E} \left[\tilde{X}(:,j) \tilde{X}(:,j)^T \right] = W \Phi W^T + \sigma_N^2 I,$$

where $\Phi = \mathbb{E} [H(:,j)H(:,j)^T]$. We have the following lemma.

Lemma 3.5. *Under Assumption 2, the matrix $\Phi = \mathbb{E} [H(:,j)H(:,j)^T]$ is given by*

$$(3.2) \quad \Phi = \frac{1}{\rho_0(\rho_0 + 1)} (D + \rho \rho^T),$$

where $D = \text{Diag}(\rho_1, \dots, \rho_r)$ and $\rho_0 = \sum_{i=1}^r \rho_i$. Also, the largest and smallest eigenvalues of Φ are bounded by

$$\lambda_{\max}(\Phi) \leq u := \frac{\rho_1 + \|\rho\|_2^2}{\rho_0(\rho_0 + 1)} \quad \text{and} \quad \lambda_{\min}(\Phi) \geq \ell := \frac{\rho_r}{\rho_0(\rho_0 + 1)}.$$

Proof. It is known that for a random vector $x \in \mathbb{R}^r$ following a Dirichlet distribution of parameter ρ , its means and covariances are given, respectively, by

$$\mathbb{E}[x_i] = \frac{\rho_i}{\rho_0} \quad \text{and} \quad \text{cov}[x_i, x_j] = \begin{cases} -\frac{\rho_i \rho_j}{\rho_0^2(\rho_0 + 1)} & \text{if } i \neq j, \\ \frac{\rho_i(\rho_0 - \rho_i)}{\rho_0^2(\rho_0 + 1)} & \text{if } i = j. \end{cases}$$

From the above results, we get

$$\Phi_{ii} = \mathbb{E}[x_i^2] = \left(\frac{\rho_i}{\rho_0} \right)^2 + \frac{\rho_i(\rho_0 - \rho_i)}{\rho_0^2(\rho_0 + 1)} = \frac{\rho_i^2 + \rho_i}{\rho_0(\rho_0 + 1)}$$

and, for $i \neq j$,

$$\Phi_{ij} = \mathbb{E}[x_i x_j] = \frac{\rho_i \rho_j}{\rho_0^2} - \frac{\rho_i \rho_j}{\rho_0^2(\rho_0 + 1)} = \frac{\rho_i \rho_j}{\rho_0(\rho_0 + 1)},$$

which lead to (3.2). The bounds on the eigenvalues follow from the fact that for any $A, B \in \mathbb{S}^r$, $\lambda_{\max}(A + B) \leq \lambda_{\max}(A) + \lambda_{\max}(B)$ and $\lambda_{\min}(A + B) \geq \lambda_{\min}(A) + \lambda_{\min}(B)$. ■

From Lemma 3.5, we deduce the following result.

Theorem 3.6. *Consider preconditioning via prewhitening. Under Assumption 2, the condition number of QW is bounded by*

$$\kappa(QW) \leq \kappa(W) \sqrt{\frac{u \sigma_{\min}^2(W) + \sigma_N^2}{\ell \sigma_{\max}^2(W) + \sigma_N^2}},$$

where

$$u = \frac{\rho_1 + \|\rho\|_2^2}{\rho_0(\rho_0 + 1)} \quad \text{and} \quad \ell = \frac{\rho_r}{\rho_0(\rho_0 + 1)}.$$

Proof. By Lemma 3.5, we have that

$$\ell WW^T + \sigma_N^2 I \preceq W\Phi W^T + \sigma_N^2 I \preceq u WW^T + \sigma_N^2 I.$$

Hence we have that $(u WW^T + \sigma_N^2 I)^{-1} \preceq (W\Phi W^T + \sigma_N^2 I)^{-1} \preceq (\ell WW^T + \sigma_N^2 I)^{-1}$. Consider $W^T Q^T Q W = W^T (W\Phi W^T + \sigma_N^2 I)^{-1} W$. Letting $W = U\Sigma V^T$ be the SVD of W , we obtain

$$\begin{aligned} W^T Q^T Q W &\preceq W^T (\ell WW^T + \sigma_N^2 I)^{-1} W \\ &= V\Sigma U^T (U(\ell\Sigma^2 + \sigma_N^2 I)U^T)^{-1} U\Sigma V^T \\ &= V\Sigma (\ell\Sigma^2 + \sigma_N^2 I)^{-1} \Sigma V^T \\ &= V \begin{pmatrix} \ddots & & & \\ & \frac{\sigma_i^2(W)}{\ell\sigma_i^2(W) + \sigma_N^2} & & \\ & & \ddots & \\ & & & \ddots \end{pmatrix} V^T. \end{aligned}$$

Hence $\frac{\sigma_{\max}^2(W)}{\ell\sigma_{\max}^2(W) + \sigma_N^2}$ is an upper bound for the largest eigenvalue of $W^T Q^T Q W$. Using the same trick, we obtain $\frac{\sigma_{\min}^2(W)}{u\sigma_{\min}^2(W) + \sigma_N^2}$ as a lower bound for the smallest eigenvalue of $W^T Q^T Q W$, which gives the result. \blacksquare

Combining Corollary 1.2 with Theorem 3.6 implies the robustness of prewhitening combined with SPA under the aforementioned generative model. It is particularly interesting to observe that, assuming $\sigma_{\min}(W) \gg \sigma_N$, we have

$$(3.3) \quad \kappa(QW) \lesssim \kappa(W) \sqrt{\frac{u\sigma_{\min}^2(W)}{\ell\sigma_{\max}^2(W)}} = \sqrt{\frac{u}{\ell}} = \sqrt{\frac{\rho_1 + \|\rho\|_2^2}{\rho_r}}.$$

As can be seen, the approximate bound above does not depend on the conditioning of W —which is appealing when we plug it into Corollary 1.2 to obtain its provable SPA error bound. That said, one should note that ρ , which characterizes how the abundances are spread, plays a role. To get more insight, consider again the symmetric distribution case $\beta := \rho_1 = \dots = \rho_r$. Equation (3.3) reduces to

$$\kappa(QW) \lesssim \sqrt{\frac{\beta + r\beta^2}{\beta}} = \sqrt{1 + r\beta}.$$

We see that fixing r , a smaller (respectively, larger) β implies an improved (respectively, degraded) bound—which is quite natural since β controls the concentration of data points around the vertices (or pure pixels). It is also interesting to look at the asymmetric distribution case. Specifically, consider an extreme case where we fix $\rho_1, \dots, \rho_{r-1}$ and scale ρ_r to a very

small value. Physically, this means one endmember is present in very small proportions in the data set—a scenario reminiscent of the worst-case scenario identified by the tight robustness analysis in the last subsection. Then, from (3.3), one can see that the bound worsens as ρ_r decreases. In fact, as ρ_r goes to zero, the r th endmember progressively disappears from the data set and hence cannot be recovered: for $\rho_r \rightarrow 0$, (3.3) becomes unbounded, which matches the result in Theorem 3.4 for $n \rightarrow \infty$ (in a worst-case scenario).

To conclude, SPA preconditioned by prewhitening can yield good performance if the abundances are more uniformly spread and a good population of them is close to the pure pixels. On the other hand, one will expect deteriorated performance if one of the endmembers exhibits little contributions in the data set or if most of the pixels are heavily mixed.

4. SPA-based heuristic preconditioning. As discussed previously, the intuition behind designing a good preconditioner is to find the left inverse W^\dagger of W in the ideal case or to efficiently approximate W^\dagger in practice. This reminds us that the original SPA (or SPA without preconditioning) can extract W exactly in the noiseless case and approximately in the noisy case (Theorem 1.1). Hence, a possible heuristic, which has been explored in [17], is as follows:

1. Identify approximately the columns of W among the columns of \tilde{X} using SPA; that is, identify an index set \mathcal{K} such that $W \approx \tilde{X}(:, \mathcal{K})$ (note that other pure-pixel search algorithms could be used).

2. Compute $\tilde{X}(:, \mathcal{K})^\dagger = (\tilde{X}(:, \mathcal{K})^T \tilde{X}(:, \mathcal{K}))^{-1} \tilde{X}(:, \mathcal{K})^T$ (this is prewhitening of $\tilde{X}(:, \mathcal{K})$).

The computational cost is that of SPA, which requires $2mnr + \mathcal{O}(mr^2)$ operations [18], plus that of the SVD of $\tilde{X}(:, \mathcal{K})$, which requires $\mathcal{O}(mr^2)$ operations. Hence, this SPA-based preconditioning is simple and computationally very efficient. Our interest with the SPA-based preconditioning is fundamental. We will draw a connection between the SPA-based preconditioning and the (arguably ideal) SDP-based preconditioning. Then, a robustness analysis will be given.

Note that, for $m = r$, the preconditioning is given by $\tilde{X}(:, \mathcal{K})^{-1}$, while $(\tilde{X}(:, \mathcal{K})\tilde{X}(:, \mathcal{K})^T)^{-1}$ is the optimal solution of

$$(4.1) \quad P^* = \underset{P \in \mathbb{S}_+^r}{\operatorname{argmax}} \det(P) \text{ such that } \tilde{x}_j^T P \tilde{x}_j \leq 1 \quad \forall j \in \mathcal{K};$$

see [19, Thm. 4] (this also follows from Lemma 3.1). Hence our heuristic can be seen as a relaxation of the SDP (1.1) where we have selected a subset of the constraints using SPA. Moreover, by letting A^* be the optimal solution of (1.1), we have

$$\det(P^*) \geq \det(A^*).$$

Therefore, we can easily provide an a posteriori robustness analysis, observing that

$$\det(P^*) \geq \det(A^*) \geq \det\left(\frac{P^*}{\max_j \tilde{x}_j^T P^* \tilde{x}_j}\right)$$

since $\frac{P^*}{\max_j \tilde{x}_j^T P^* \tilde{x}_j}$ is a feasible point of (1.1). Therefore, by denoting $\beta = \max_j \tilde{x}_j^T P^* \tilde{x}_j$, $\frac{1}{\beta} P^*$ is a $\frac{1}{\beta r}$ -approximate solution of SDP (1.1), and we can apply Theorem 2.6.

Remark 1. Note that the active set method for (1.1) proposed in [19] implicitly uses the above observation. In fact, the set of initial constraints was selected using SPA and updated

by adding the most violated constraints at each step (that is, the constraints corresponding to the largest $\tilde{x}_j^T P^* \tilde{x}_j$).

We have observed that extracting more than r columns with SPA sometimes gives a better preconditioning. Intuitively, extracting more columns allows one to better assess the way the columns of \tilde{X} are spread in space: at the limit, if all columns are extracted, this is exactly Algorithm **NF-PW**. Hence we have added a parameter $r \leq p \leq \min(m, n)$; see Algorithm **SPA-Prec**. Note that SPA cannot extract more than $\text{rank}(\tilde{X})$ indices. Therefore, in the noiseless case ($\text{rank}(\tilde{X}) = r$), the SPA-based preconditioning performs perfectly for any $p \geq r$.

Algorithm SPA-Prec – SPA-based Preconditioning [17].

Input: Matrix $\tilde{X} = X + N$ with X satisfying Assumption 1, rank r , parameter $r \leq p \leq \min(m, n)$.

Output: Preconditioner Q .

- 1: $\mathcal{K} = \text{SPA}(\tilde{X}, p)$.
 - 2: $Q = \text{NF-PW}(\tilde{X}(:, \mathcal{K}), r)$.
-

SPA-Prec can be used to make pure-pixel search algorithms more robust, e.g., SPA. It is kind of surprising: one can use SPA to precondition SPA and make it more robust.

4.1. Robustness analysis. We can provide the following robustness results for SPA preconditioned with SPA.

Theorem 4.1. *Let $\tilde{X} = X + N$, where X satisfies Assumption 1 with $m = r$, W has full rank, and the noise N satisfies $\max_j \|N(:, j)\|_2 \leq \epsilon$. If $\epsilon \leq \mathcal{O}\left(\frac{\sigma_{\min}(W)}{\sqrt{r\kappa^2(W)}}\right)$, then SPA-based preconditioned SPA identifies the columns of W up to error $\mathcal{O}(\epsilon\kappa(W))$.*

Proof. Let us denote $\tilde{W} = \tilde{X}(:, \mathcal{K}) \in \mathbb{R}^{r \times r}$ to be the matrix extracted by SPA so that the SPA-based preconditioning is given by $Q = \tilde{W}^{-1}$. Hence, using Corollary 1.2, it remains to prove that

$$\kappa(QW) = \kappa(\tilde{W}^{-1}W) = \kappa(W^{-1}\tilde{W}) = \mathcal{O}(1).$$

Let us assume without loss of generality that the columns of \tilde{W} are properly permuted (this does not affect the preconditioning) so that, by Theorem 1.1, we have $E = W - \tilde{W}$, where

$$\max_j \|E(:, j)\|_2 \leq \mathcal{O}(\epsilon\kappa(W)^2) \leq \mathcal{O}\left(\frac{\sigma_{\min}(W)}{\sqrt{r}}\right).$$

This implies that $\|E\|_2 \leq \mathcal{O}(\sigma_{\min}(W))$. By denoting (U, Σ_W, V) to be the SVD of W , we have

$$W^{-1}\tilde{W} = V\Sigma_W^{-1}U^T(U\Sigma_W V^T + E) = I_r + V\Sigma_W^{-1}U^T E.$$

Denoting $A = V\Sigma_W^{-1}U^T E$, we also have

$$\kappa(W^{-1}\tilde{W}) \leq \frac{1 + \sigma_{\max}(A)}{1 - \sigma_{\max}(A)},$$

where $\|V\Sigma_W^{-1}U^T E\|_2 \leq \|\Sigma_W^{-1}\|_2 \|E\|_2 = \sigma_{\min}(W)^{-1} \|E\|_2 \leq \mathcal{O}(1)$, which gives the result. The above bound on the condition number follows from the singular value perturbation theorem; see, e.g., [20, Cor. 8.6.2], which states that, for any square matrix $A = B - B'$, we have $|\sigma_i(B) - \sigma_i(B')| \leq \sigma_{\max}(A)$ for all i . ■

It is interesting to notice the following.

- SPA-based preconditioned SPA improves the error bound of SPA by a factor $\kappa(W)$.
- The theoretical result for SPA-based preconditioned SPA (Theorem 4.1) does not allow higher noise levels than SPA. However, in practice, it will allow much higher noise levels; see the numerical experiments in section 5.
- SPA-based preconditioned SPA has the same robustness as postprocessed SPA [3] (see also [19] for a discussion), although SPA-based preconditioned SPA is computationally slightly cheaper (postprocessed SPA requires r orthogonal projections onto $(r - 1)$ -dimensional subspaces). Moreover, postprocessed SPA was shown to perform only slightly better than SPA [19], while SPA-based preconditioned SPA will outperform SPA (in particular, this applies to the synthetic data sets described in section 5.2).
- The procedure can potentially be used recursively, that is, use the solution obtained by SPA preconditioned with SPA to precondition SPA. However, we have not observed significant improvement in doing so, and the error bound that can be derived is asymptotically the same as for a single-pass SPA-based preconditioning. In fact, Theorem 4.1 can be easily adapted: the only difference in the proof is that the upper bound for $\|E\|_2$ would be better, from $\mathcal{O}(\sigma_{\min}(W))$ to $\mathcal{O}(\frac{\sigma_{\min}(W)}{\kappa(W)})$, which does not influence $\kappa(W^{-1}\tilde{W})$ being in $\mathcal{O}(1)$.

5. Numerical experiments. In this section, we compare the following algorithms:

- *SPA*. The successive projection algorithm; see Algorithm SPA.
- *SDP-SPA*. Algorithm SDP-Prec + SPA.
- *PW-SPA*. Algorithm NF-PW + SPA.
- *SPA-SPA*. Algorithm SPA-Prec ($p = r$) + SPA.
- *VCA*. Vertex component analysis (VCA) [30], available from <http://www.lx.it.pt/~bioucas>.
- *XRAY*. Fast conical hull algorithm, “max” variant [24].

The MATLAB code is available from <https://sites.google.com/site/nicolasgillis/>. For SPA, VCA, and SDP-SPA, we used the original code of the authors. (Note that the SDPs are solved via an active set method where the SDP subproblems are solved with interior point methods using the parser-solver CVX [21].) For XRAY, we used the code from [16] (as it was not made available by the authors). All tests are performed using MATLAB on a laptop with Intel CORE i5-3210M 2.5GHz CPU and with 6GB RAM.

Table 1 summarizes the theoretical results for SPA and its preconditioned variants.

5.1. Two-by-three near-separable matrix. In this subsection, we illustrate the effectiveness of the preconditionings on a small near-separable matrix: let

$$W = \begin{pmatrix} k+1 & k \\ k & k+1 \end{pmatrix}$$

Table 1

Robustness and computational cost of the different preconditionings. The best performances are highlighted in bold. The algorithm α -SDP-SPA refers to SPA preconditioned with an α -approximate solution of (1.1) (see section 2), while PW-SPA + model refers to PW-SPA where the generative model is Dirichlet with parameter ρ plus Gaussian noise N where $\sigma_{\max}(N) \ll \sigma_{\min}(W)$ (see section 3.4).

	Noise level (ϵ) $= \mathcal{O}\left(\frac{\sigma_{\min}(W)}{\sqrt{r}}\right)$	Error $= \mathcal{O}(\epsilon\kappa(W))$	Comp. cost	Reference
SPA	$\kappa^{-2}(W)$	$\kappa(W)$	$\mathbf{O}(\mathbf{mnr})$	[18, Thm. 3]
SDP-SPA	\mathbf{r}^{-1}	$\mathbf{1}$	$\Omega(mnr^2)$	[19, Thm. 2.9]
α -SDP-SPA	$\min(r^{-1}, \alpha^{3/2})$	$\alpha^{-3/2}$	$\Omega(mnr^2)$	Theorem 2.6
PW-SPA	$n^{-3/2}$	$n^{3/2}$	$\mathcal{O}(mnr^2)$	Theorem 3.4
PW-SPA + model	$\left(\frac{\rho_1 + \ \rho\ _2^2}{\rho_r}\right)^{-3/2}$	$\left(\frac{\rho_1 + \ \rho\ _2^2}{\rho_r}\right)^{3/2}$	$\mathcal{O}(mnr^2)$	Theorem 3.6
SPA-SPA	$\kappa^{-2}(W)$	$\mathbf{1}$	$\mathbf{O}(\mathbf{mnr})$	Theorem 4.1

for some parameter $k \geq 0$. We have that $\sigma_{\min}(W) = 1$ and $\sigma_{\max}(W) = 2k + 1$, and hence $\kappa(W) = 2k + 1$. Let us also take

$$(5.1) \quad H = \begin{pmatrix} 1 & 0 & 0.5 \\ 0 & 1 & 0.5 \end{pmatrix}, \quad \tilde{X} = WH + N,$$

with $N = \delta [-W(:, 1), -W(:, 2), WH(:, 3)]$. Under the above setup, the following phenomena can be shown: for $\delta \geq \frac{1}{8k^2}$, we have $\|\tilde{X}(:, 1)\|_2^2 = \|\tilde{X}(:, 2)\|_2^2 < \|\tilde{X}(:, 3)\|_2^2$. Subsequently, SPA will extract $\tilde{X}(:, 3)$ as an endmember estimate, which is wrong. To show this, note that

$$\|\tilde{X}(:, 1)\|_2 = \|\tilde{X}(:, 2)\|_2 = (1 - \delta)\sqrt{2k^2 + 2k + 1}$$

while

$$\|\tilde{X}(:, 3)\|_2 = (1 + \delta)\sqrt{2k^2 + 2k + \frac{1}{2}}.$$

The condition $\|\tilde{X}(:, 1)\|_2^2 = \|\tilde{X}(:, 2)\|_2^2 < \|\tilde{X}(:, 3)\|_2^2$ happens when

$$\delta \geq \frac{1}{8k^2} > \frac{\sqrt{2k^2 + 2k + 1} - \sqrt{2k^2 + 2k + \frac{1}{2}}}{\sqrt{2k^2 + 2k + 1} + \sqrt{2k^2 + 2k + \frac{1}{2}}},$$

where the second inequality is obtained via

$$\sqrt{2k^2 + 2k + 1} + \sqrt{2k^2 + 2k + \frac{1}{2}} > 2k$$

and

$$\sqrt{2k^2 + 2k + 1} - \sqrt{2k^2 + 2k + \frac{1}{2}} < \frac{1}{4k}.$$

(The second inequality can be obtained by multiplying the left- and right-hand sides by $\sqrt{2k^2 + 2k + 1} + \sqrt{2k^2 + 2k + \frac{1}{2}}$.) Also, it can be shown that, for any k , SDP-SPA will extract

the columns of W correctly, with error proportional to $\mathcal{O}(\delta k)$ for $\delta \leq \mathcal{O}(1)$. Figure 1 displays the fraction of columns of W properly identified by the different algorithms for different value of δ : on the left for $k = 10$ and on the right for $k = 1000$.

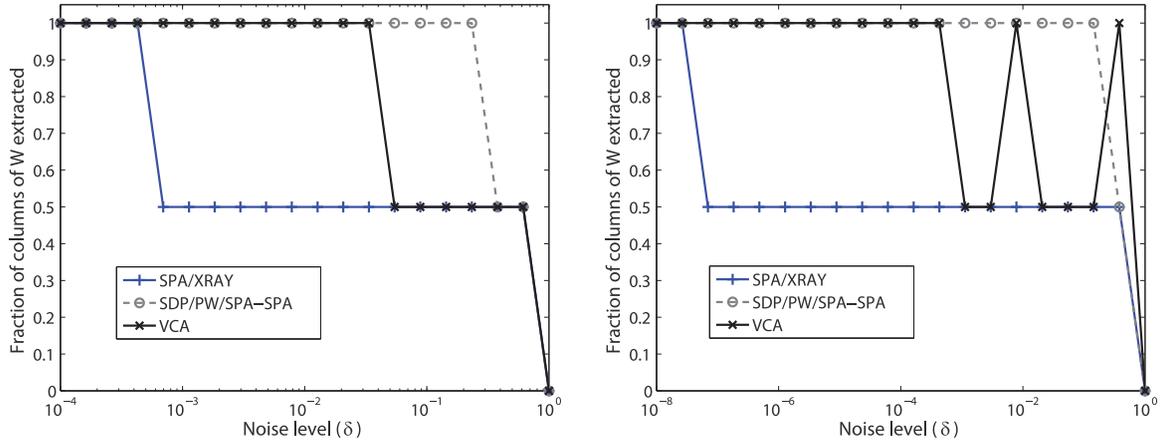


Figure 1. Comparison of the different near-separable NMF algorithms on the matrix from (5.1) with $k = 10$ (left) and $k = 1000$ (right).

As explained above, SPA fails to properly identify the two columns of W for any $\delta \leq \mathcal{O}(k^{-2})$, while SDP-SPA works perfectly for $\delta \leq \mathcal{O}(1)$. It turns out that all preconditioned variants perform the same, while XRAY performs the same as SPA. VCA is not deterministic, and different runs lead to different outputs. In fact, potentially any column of \tilde{X} can be extracted by VCA for $\delta > 0$.

5.2. Middle point experiment. In this subsection, we use the so-called middle point experiment from [18], with $m = 40$, $r = 20$, and $n = 210$. The input matrix satisfies Assumption 1, where each entry of W is generated uniformly at random in $[0,1]$ and H' contains only two nonzero entries equal to 0.5 (hence all data points are in the middle of two columns of W). The noise moves the middle points toward the outside of the convex hull of the columns of W with $N(:, j) = \delta (X(:, j) - \bar{w})$, where $\bar{w} = \frac{1}{r} \sum_{k=1}^r W(:, k)$ and δ is the noise parameter; see [18] for more details.

For each noise level (from 0 to 0.6 with step 0.01), we generate 25 such matrices, and Figure 2 and Table 2 report the numerical results.

We observe that SPA-SPA is able to improve the performance of SPA significantly: for example, for the noise level $\delta = 0.4$, SPA correctly identifies about 20% of the columns of W , while SPA-SPA identifies about 95%. Note that SPA-SPA is only slightly faster than PW-SPA because (i) m is not much larger than r , and (2) as opposed to SPA-SPA, PW-SPA actually does not need to compute the product $Q\tilde{X}$ where Q is the preconditioning since $Q\tilde{X} = V_r^T$. For large m and n , SPA-SPA will be much faster (see the next section for an example). Note also that PW-SPA performs very well (in fact, as well as SDP-SPA) because the data points are well spread in the convex hull of the columns of W , and hence $\kappa(H)$ is close to one (in fact, it is equal to 1.38, while the average value of $\kappa(W)$ is around 22.5).

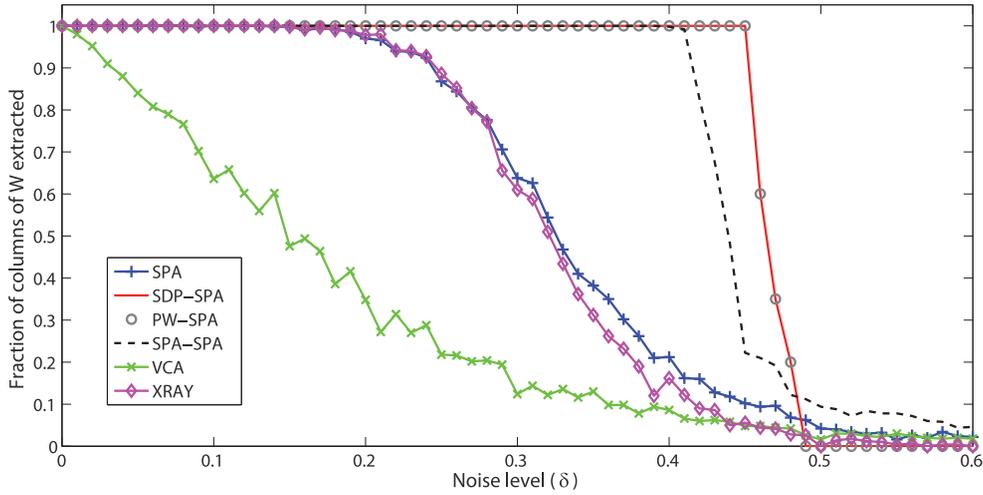


Figure 2. Comparison of the different near-separable NMF algorithms on the middle point experiment.

Table 2

Robustness (that is, the largest value of δ for which all columns of W are correctly identified) and total running time in seconds of the different near-separable NMF algorithms.

	Robustness	Total time (s.)
SPA	0.08	4
SDP-SPA	0.45	3508
PW-SPA	0.45	34
SPA-SPA	0.39	31
VCA	0	841
XRAY	0.18	743

5.3. Hubble telescope. We use the simulated noisy Hubble telescope hyperspectral image from [33] (with $m = 100$ and $n = 16384$) constituted of eight endmembers (see Figure 3). Table 3 reports the running time and the mean-removed spectral angle (MRSA) between the true endmembers (of the clean image) and the extracted endmembers. Given two spectral signatures, $x, y \in \mathbb{R}^m$, the MRSA is defined as

$$\phi(x, y) = \frac{100}{\pi} \arccos \left(\frac{(x - \bar{x})^T (y - \bar{y})}{\|x - \bar{x}\|_2 \|y - \bar{y}\|_2} \right) \in [0, 100].$$

Figure 3 displays the abundance maps corresponding to the extracted columns of W .

All preconditioned variants are able to identify the eight materials properly, as opposed to the original SPA. SPA-SPA performs slightly better than the other preconditioned variants while being the fastest. We do not show the results of VCA and XRAY, as they perform very poorly [19].

6. Conclusion and further research. In this paper, we analyzed several preconditionings for making pure-pixel search algorithms more robust to noise: an approximate SDP,

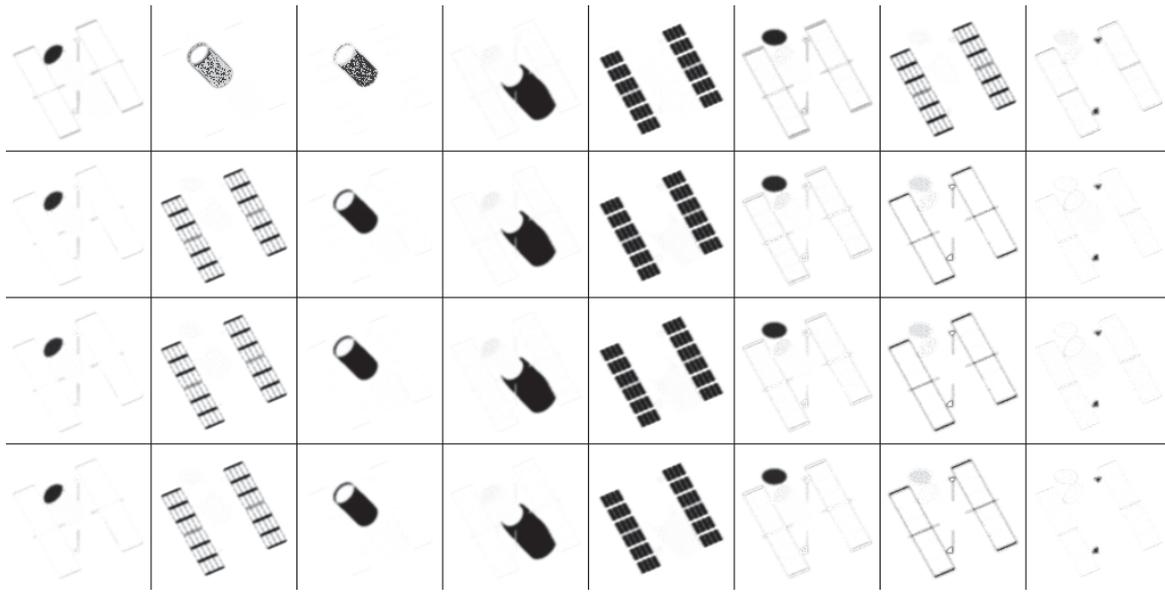


Figure 3. The abundance maps corresponding to the endmembers extracted by the different algorithms. From top to bottom: SPA, SDP-SPA, PW-SPA, SPA-SPA. From left to right: honeycomb side, copper stripping, green glue, aluminum, solar cell, honeycomb top, black rubber edge, and bolts.

Table 3

MRSA of the identified endmembers with the true endmembers and running time in seconds of the different preconditioned SPA algorithms.

	SPA	SDP-SPA	PW-SPA	SPA-SPA
Hon. side	6.51	6.94	6.94	6.15
Cop. strip.	26.83	7.46	7.44	7.44
Green glue	2.09	2.03	2.03	2.03
Aluminum	1.71	1.80	1.80	1.80
Solar cell	4.96	5.48	5.48	4.96
Hon. top	2.34	2.30	2.30	2.30
Black edge	27.09	13.16	13.16	13.13
Bolts	2.65	2.65	2.65	2.70
Average	9.27	5.23	5.23	5.06
Time (s.)	0.05	4.74	2.18	0.37

prewhitening, and a simple and fast yet effective SPA-based preconditioning. The analyses revealed that these preconditionings, which aim at low-complexity implementation and are suboptimal compared to the ideal SDP preconditioning, actually have provably good error bounds on pure-pixel identification performance.

Further research includes the following tasks:

- Evaluate the preconditionings on real-world hyperspectral images. We have performed preliminary numerical experiments on real-world hyperspectral images and did not observe significant advantages when using the different preconditionings. A plausible explanation is that the noise level in such images is usually rather large (in particular

larger than the bounds derived in the theorems) and these images contain outliers. Hence, to make preconditionings effective in such conditions, some preprocessing of the data would be necessary, in particular outlier identification, since pure-pixel search algorithms are usually very sensitive to outliers (e.g., VCA, SPA, and XRAY).

- Use the preconditioning to enhance other blind HU algorithms, for example algorithms which do not require the pure-pixel assumption to hold; see, e.g., [26, 9, 6].
- Analyze theoretically and practically the influence of preconditioning on other pure-pixel search algorithms. For example, the results of this paper directly apply to the successive nonnegative projection algorithm (SNPA), which is more robust and applies to a broader class of matrices (W does not need to be full rank) than SPA [15].

Appendix A. Proof of Lemma 2.4. We have to prove that

$$\kappa^* = \frac{1 + \sqrt{1 - \gamma \left(\frac{r}{\beta}\right)^r}}{1 - \sqrt{1 - \gamma \left(\frac{r}{\beta}\right)^r}}$$

is the optimal value of $\max_{\lambda \in \mathbb{R}^r} \frac{\lambda_1}{\lambda_r}$ such that $\sum_i \lambda_i \leq \beta$, $\prod_i \lambda_i \geq \gamma$, and $\lambda_1 \geq \dots \geq \lambda_r \geq 0$, where $\beta \geq r$ and $0 < \gamma \leq 1$. Note first that the problem is feasible taking $\lambda_i = 1$ for all i .

At optimality, the constraint $\sum_i \lambda_i \leq \beta$ must be active (otherwise λ_1 can be increased to generate a strictly better solution), the constraint $\prod_i \lambda_i \geq \gamma$ must also be active (otherwise λ_r can be decreased to obtain a strictly better solution), and $\lambda_i > 0$ for all i (otherwise the solution is infeasible since $\gamma > 0$).

The feasible domain is compact, and the objective function is continuous and bounded above: in fact, $\lambda_i \leq \beta$ and $\lambda_i \geq \frac{\gamma}{\beta^{r-1}}$ for all i , and hence $\kappa^* \leq \frac{\beta^r}{\gamma}$. Therefore, the maximum must be attained (extreme value theorem). Let λ^* be an optimal solution.

For $r = 2$, λ^* must satisfy $\lambda_1^* + \lambda_2^* = \beta$, $\lambda_1^* \lambda_2^* = \gamma$, and $\lambda_1^* \geq \lambda_2^*$, and hence $\lambda_1^* = \frac{\beta}{2} \left(1 + \sqrt{1 - \gamma \frac{4}{\beta^2}}\right)$ and $\lambda_2^* = \frac{\beta}{2} \left(1 - \sqrt{1 - \gamma \frac{4}{\beta^2}}\right)$, which gives the result.

For $r \geq 3$, let us show that $\lambda_i^* = \lambda_{i+1}^*$ for all $2 \leq i \leq r - 2$. Assume $\lambda_i^* > \lambda_{i+1}^*$ for some $2 \leq i \leq r - 2$. Replacing λ_i^* and λ_{i+1}^* by their average will keep their sum constant while strictly increasing their product; hence this generates another optimal solution, a contradiction since the constraint $\prod_i \lambda_i \geq \gamma$ must be active at optimality. Therefore, the above optimization problem is equivalent to

$$(A.1) \quad \begin{aligned} \kappa^* = \max_{x \in \mathbb{R}^3} \quad & \frac{x_1}{x_3} \text{ such that } x_1 + (r - 2)x_2 + x_3 = \beta, \\ & x_1 x_2^{r-2} x_3 = \gamma, \text{ and} \\ & x_1 \geq x_2 \geq x_3 \geq 0. \end{aligned}$$

Let us consider a relaxation of the above problem by dropping the constraint $x_1 \geq x_2 \geq x_3 \geq 0$; it will be shown that the solution of the relaxed problem satisfies the constraints automatically.

The first-order optimality conditions of the relaxed problem are given by

$$\begin{aligned}\frac{1}{x_3} &= \lambda + \mu x_2^{r-2} x_3 = \lambda + \mu \frac{\gamma}{x_1}, \\ 0 &= \lambda(r-2) + \mu(r-2)x_2^{r-3} x_1 x_3 = (r-2) \left(\lambda + \mu \frac{\gamma}{x_2} \right), \\ \frac{-x_1}{x_3^2} &= \lambda + \mu x_2^{r-2} x_1 = \lambda + \mu \frac{\gamma}{x_3},\end{aligned}$$

where λ and μ are the Lagrangian multipliers for the first and second constraints of (A.1), respectively. Multiplying the first equality by x_1 and the third by x_3 and summing them up gives

$$x_1 + x_3 = \frac{-2\mu\gamma}{\lambda}.$$

Multiplying the second equality by x_2 gives $x_2 = \frac{-\mu\gamma}{\lambda}$, and hence $x_2 = \frac{x_1+x_3}{2}$. This gives $x_1 + (r-2)\frac{x_1+x_3}{2} + x_3 = \beta$, and hence $x_1 + x_3 = 2\frac{\beta}{r}$, which combined with $x_1 \left(\frac{x_1+x_3}{2}\right)^{r-2} x_3 = \gamma$ gives $x_1 x_3 = \gamma \left(\frac{r}{\beta}\right)^{r-2}$. Finally, the solution is given by

$$x_1^* = \frac{\beta}{r} \left(1 + \sqrt{1 - \gamma \left(\frac{r}{\beta}\right)^r} \right), \quad x_3^* = \frac{\beta}{r} \left(1 - \sqrt{1 - \gamma \left(\frac{r}{\beta}\right)^r} \right),$$

and $x_2^* = \frac{x_1^*+x_3^*}{2}$. It can be seen that for $\beta \geq \gamma$ and $0 < \gamma \leq 1$, we have $x_1^* \geq x_2^* \geq x_3^* \geq 0$, which satisfies the third constraint of (A.1) automatically. Hence, the (x_1^*, x_2^*, x_3^*) above is the optimal solution of (A.1).

Acknowledgment. The authors would like to thank the reviewers for their insightful comments which helped improve the paper.

REFERENCES

- [1] S. D. AHIPASOGLU, P. SUN, AND M. J. TODD, *Linear convergence of a modified Frank-Wolfe algorithm for computing minimum-volume enclosing ellipsoids*, *Optim. Methods Softw.*, 23 (2008), pp. 5–19.
- [2] U. M. C. ARAÚJO, B. T. C. SALDANHA, R. K. H. GALVÃO, T. YONEYAMA, H. C. CHAME, AND V. VISANI, *The successive projections algorithm for variable selection in spectroscopic multicomponent analysis*, *Chemometr. Intell. Lab. Syst.*, 57 (2001), pp. 65–73.
- [3] S. ARORA, R. GE, Y. HALPERN, D. MIMNO, A. MOITRA, D. SONTAG, Y. WU, AND M. ZHU, *A practical algorithm for topic modeling with provable guarantees*, in *Proceedings of the 30th International Conference on Machine Learning (ICML '13)*, Vol. 28, 2013, pp. 280–288.
- [4] S. ARORA, R. GE, R. KANNAN, AND A. MOITRA, *Computing a nonnegative matrix factorization – provably*, in *Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC '12)*, 2012, pp. 145–162.
- [5] J. BATSON, D. A. SPIELMAN, AND N. SRIVASTAVA, *Twice-Ramanujan sparsifiers*, *SIAM J. Comput.*, 41 (2012), pp. 1704–1721.
- [6] J. BIUCAS-DIAS, *A variable splitting augmented Lagrangian approach to linear spectral unmixing*, in *Proceedings of the 1st IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, Grenoble, France, 2009.

- [7] J. M. BIOCAS-DIAS, A. PLAZA, N. DOBIGEON, M. PARENTE, Q. DU, P. GADER, AND J. CHANUSSOT, *Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches*, IEEE J. Sel. Topics Appl. Earth Observ. in Remote Sens., 5 (2012), pp. 354–379.
- [8] S. BOYD AND L. VANDENBERGHE, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.
- [9] T.-H. CHAN, C.-Y. CHI, Y.-M. HUANG, AND W.-K. MA, *A convex analysis based minimum-volume enclosing simplex algorithm for hyperspectral unmixing*, IEEE Trans. Signal Process., 57 (2009), pp. 4418–4432.
- [10] T.-H. CHAN, W.-K. MA, C.-Y. CHI, AND Y. WANG, *A convex analysis framework for blind separation of non-negative sources*, IEEE Trans. Signal Process., 56 (2008), pp. 5120–5134.
- [11] Y. CHEN, J. MAIRAL, AND Z. HARCHAOU, *Fast and robust archetypal analysis for representation learning*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR' 14), 2014, pp. 1478–1485.
- [12] P. COMON AND C. JUTTEN, *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, Elsevier, Amsterdam, 2010.
- [13] A. CUTLER AND L. BREIMAN, *Archetypal analysis*, Technometrics, 36 (1994), pp. 338–347.
- [14] E. ESSER, M. MOLLER, S. OSHER, G. SAPIRO, AND J. XIN, *A convex model for nonnegative matrix factorization and dimensionality reduction on physical space*, IEEE Trans. Image Process., 21 (2012), pp. 3239–3252.
- [15] N. GILLIS, *Successive nonnegative projection algorithm for robust nonnegative blind source separation*, SIAM J. Imaging Sci., 7 (2014), pp. 1420–1450.
- [16] N. GILLIS AND R. LUCE, *Robust near-separable nonnegative matrix factorization using linear optimization*, J. Mach. Learn. Res., 15 (2014), pp. 1249–1280.
- [17] N. GILLIS AND W.-K. MA, *Enhancing pure-pixel identification performance via preconditioning*, in Proceedings of the 6th IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), Lausanne, Switzerland, 2014, to appear.
- [18] N. GILLIS AND S. A. VAVASIS, *Fast and robust recursive algorithms for separable nonnegative matrix factorization*, IEEE Trans. Pattern Anal. Mach. Intell., 36 (2014), pp. 698–714.
- [19] N. GILLIS AND S. A. VAVASIS, *Semidefinite programming based preconditioning for more robust near-separable nonnegative matrix factorization*, SIAM J. Optim., 25 (2015), pp. 677–698.
- [20] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computation*, 3rd ed., The Johns Hopkins University Press, Baltimore, MD, 1996.
- [21] M. GRANT AND S. BOYD, *CVX: MATLAB Software for Disciplined Convex Programming, Version 1.21*, <http://cvxr.com/cvx/>, April 2011.
- [22] F. JOHN, *Extremum Problems with Inequalities as Subsidiary Conditions*, Fritz John: Collected Papers, Vol. 2, J. Moser, ed., Birkhäuser, Boston, 1985, pp. 543–560; first published in 1948.
- [23] L. G. KHACHIYAN, *Rounding of polytopes in the real number model of computation*, Math. Oper. Res., 21 (1996), pp. 307–320.
- [24] A. KUMAR, V. SINDHWANI, AND P. KAMBADUR, *Fast conical hull algorithms for near-separable non-negative matrix factorization*, in Proceedings of the 30th International Conference on Machine Learning (ICML '13), Vol. 28, 2013, pp. 231–239.
- [25] D. D. LEE AND H. S. SEUNG, *Learning the parts of objects by nonnegative matrix factorization*, Nature, 401 (1999), pp. 788–791.
- [26] J. LI AND J. BIOCAS-DIAS, *Minimum volume simplex analysis: A fast algorithm to unmix hyperspectral data*, in Proceedings of the IEEE International Symposium on Geoscience and Remote Sensing (IGARSS), 2008.
- [27] W.-K. MA, J. M. BIOCAS-DIAS, T.-H. CHAN, N. GILLIS, P. GADER, A. PLAZA, A. AMBIKAPATHI, AND C.-Y. CHI, *A signal processing perspective on hyperspectral unmixing*, IEEE Signal Process. Mag., 31 (2014), pp. 67–81.
- [28] T. MIZUTANI, *Ellipsoidal rounding for nonnegative matrix factorization under noisy separability*, J. Mach. Learn. Res., 15 (2014), pp. 1011–1039.
- [29] A. NAOR, *Sparse quadratic forms and their geometric applications*, Astérisque, 348 (2012), pp. 189–217.
- [30] J. M. P. NASCIMENTO AND J. M. BIOCAS-DIAS, *Vertex component analysis: A fast algorithm to unmix hyperspectral data*, IEEE Trans. Geosci. Rem. Sens., 43 (2005), pp. 898–910.

- [31] J. M. P. NASCIMENTO AND J. M. BIUCAS-DIAS, *Hyperspectral unmixing based on mixtures of Dirichlet components*, IEEE Trans. Geosci. Rem. Sens., 50 (2012), pp. 863–878.
- [32] 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.
- [33] V. P. PAUCA, J. PIPER, AND R. J. PLEMMONS, *Nonnegative matrix factorization for spectral data analysis*, Linear Algebra Appl., 406 (2006), pp. 29–47.
- [34] P. SUN AND R. M. FREUND, *Computation of minimum-volume covering ellipsoids*, Oper. Res., 52 (2004), pp. 690–706.
- [35] J. YANG, D. SUN, AND K.-C. TOH, *A proximal point algorithm for log-determinant optimization with group lasso regularization*, SIAM J. Optim., 23 (2013), pp. 857–893.