Simplex-Structured Matrix Factorization: Sparsity-based Identifiability and Provably Correct Algorithms[*]

Maryam Abdolali[†] and Nicolas Gillis[†]

Abstract. In this paper, we provide novel algorithms with identifiability guarantees for simplex-structured matrix factorization (SSMF), a generalization of nonnegative matrix factorization. Current state-of-the-art algorithms that provide identifiability results for SSMF rely on the sufficiently scattered condition (SSC) which requires the data points to be well spread within the convex hull of the basis vectors. The conditions under which our proposed algorithms recover the unique decomposition is in most cases much weaker than the SSC. We only require to have d points on each facet of the convex hull of the basis vectors whose dimension is $d-1$. The key idea is based on extracting facets containing the largest number of points. We illustrate the effectiveness of our approach on synthetic data sets and hyperspectral images, showing that it outperforms state-of-the-art SSMF algorithms as it is able to handle higher noise levels, rank deficient matrices, outliers, and input data that highly violates the SSC.

Key words. simplex-structured matrix factorization, nonnegative matrix factorization, sparsity, identifiability, uniqueness, minimum volume

AMS subject classifications. 15A23, 65F50, 94A12

1. Introduction.

Extracting meaningful underlying structures that are present in high-dimensional data sets is a key problem in machine learning, data mining, and signal processing. Structured matrix factorization (SMF) is a general model for exploiting latent linear structures from data; see for example [40, 19] and the references therein. Given a factorization rank r, SMF expresses the input matrix $X \in \mathbb{R}^{m\times n}$ as the product of two matrices $W \in \mathbb{R}^{m\times r}$ and $H \in \mathbb{R}^{r\times n}$, with some restrictions on the structure of W and/or H. This paper focuses on a specific SMF model called simplex-structured matrix factorization (SSMF).

Given an m-by-n matrix X and an integer r, SSMF looks for an m-by-r matrix W whose columns are the basis vectors, and an r-by-n matrix H containing the mixing weights such that $X \approx WH$ and with the property that each column of H belongs to the unit simplex, that is, $H(:,j) \in \Delta^r = \left\{ x \in \mathbb{R}^r \;\middle|\; x \geq 0, \sum_{i=1}^r x_i = 1 \right\}$ for all j. In the exact case when $X = WH$, we have $\operatorname{conv}(X) \subseteq \operatorname{conv}(W)$ where $\operatorname{conv}(W) = \{x \mid x = Wh, h \in \Delta^n\}$, that is, each column of X belongs to the convex hull generated by the columns of W. SSMF is a generalization of nonnegative matrix factorization (NMF), an SMF problem where W and H are required to be nonnegative, while X is nonnegative as well. The main advantage of NMF over other SMFs such as the PCA/SVD is its interpretability when the factors W and H have a physical meaning; see [10, 22, 14] and the references therein. In the exact case, NMF can be formulated

[*]Submitted to the editors December 9, 2020.

Funding: The authors acknowledge the support by the European Research Council (ERC starting grant no 679515), and by the Fonds de la Recherche Scientifique - FNRS and the Fonds Wetenschappelijk Onderzoek - Vlaanderen (FWO) under EOS Project no O005318F-RG47.

[†]Department of Mathematics and Operational Research, Faculté Polytechnique, Université de Mons, Rue de Houdain 9, 7000 Mons, Belgium (maryam.abdolali@umons.ac.be, nicolas.gillis@umons.ac.be).

as an SSMF problem using a simple scaling of the columns of X and W. In fact, defining[1] D_X as the diagonal matrix with $(D_X)_{ii} = ||X(:,i)||_1$ for all i, we have

$$\underbrace{X(D_X)^{-1}}_{X'} = \underbrace{W(D_W)^{-1}}_{W'} \underbrace{D_W H(D_X)^{-1}}_{H'}.$$

Since the entries of each column of X' and W' sum to one, and since $X'(:,j) = W'H'(:,j)$ for all j, the entries of the columns of H' must also sum to one, that is, $H'(:,j) \in \Delta^r$ for all j. In fact, letting e be the vector of all ones of appropriate dimension, we have $e^\top = e^\top X' = e^\top W'H' = e^\top H'$. Note that SSMF is a constrained variant of semi-NMF which only requires the factor H to be nonnegative; see [23] and the references therein.

Applications. Let us discuss in more details two applications of SSMF: blind hyperspectral unmixing, and topic modeling; see [42] and the references therein for more applications. A hyperspectral image is a data cube that consists of hundreds of two dimensional spatial images that are acquired at different contiguous wavelengths (known as spectral bands). These images have a vast variety of applications in remote sensing, military surveillance, and environmental monitoring. Due to the limited spatial resolution of hyperspectral sensors, a pixel may be a mixture from several materials located in the captured scene. Under the linear mixing assumption, identifying the materials present in the image, known as *endmembers*, can be modeled as an SSMF problem [8, 35]. Constructing the matrix X by stacking the spectral signature of the pixels as its columns, each column of W is the spectral signature of an endmember, and each column of the matrix H represents the abundance of the endmembers in the corresponding pixel. Another application of SSMF is text mining [6, 26, 15]. Let the matrix X represent a collection of documents where the (i,j)th element indicates the frequency of the ith word in the jth document. Extracting latent topic patterns across the documents and categorizing the documents according to the extracted topics is an essential task when processing textual information. By applying SSMF on the document matrix, each column of W can be interpreted as a hidden topic, and each column of H can be regarded as the proportion of the topics discussed in the corresponding document.

Identifiability. In many applications, a crucial question about SSMF is when the factors W and H can be uniquely recovered. SSMF never has a unique solution, unless some additional constraints are imposed on the factors W and/or H. In fact, if there exists a polytope $\mathrm{conv}(W)$ containing the columns of X, then any larger polytope containing $\mathrm{conv}(W)$ leads to another solution of SSMF. Suppose X is generated by multiplying the ground truth factors W_t and H_t, where the columns of H_t belong to the unit simplex. Two crucial questions are:

1. Under what conditions are the factors W_t and H_t uniquely identifiable (up to trivial ambiguities such as permutation)?
2. Does there exist a (polynomial-time) algorithm able to recover these ground truth factors W_t and H_t?

Many works have studied these questions, leading to weaker and weaker conditions on the factors W_t and/or H_t that lead to uniqueness; see Section 2 for more details. Given that W_t is identifiable, the identifiability of H_t follows from well-known results: H_t is unique if and only if all columns of X are located on k-dimensional faces of $\mathrm{conv}(W_t)$ having exactly

[1] We assume that the columns of X and W are different from zero otherwise they can be discarded.

$k+1$ vertices [39]. When W_t is full column rank, then H_t is always unique as this condition is always met. This is the reason why the identifiability results for SSMF are focused on the identification of W_t, and we also only focus on the identifiability of W_t in this paper.

Contribution and outline of the paper. The main goal of this paper is to answer the two above questions in a novel way. In Section 2, we review the main SSMF algorithms and identifiability results. Then, the main contributions of this paper are presented in the next four sections:

1. In Section 3, we provide new identifiability conditions for SSMF, referred to as the facet-based conditions (FBC), that rely on the sparsity of H, by requiring to have $d = \text{rank}(X)$ data points on each facet[2] of $\text{conv}(W)$; see Theorem 3.4. This condition is in most cases much weaker than the current state-of-the-art identifiability conditions that rely on the data points being sufficiently spread within $\text{conv}(W)$.
2. In Section 4, we propose and study a first algorithm for SSMF, dubbed brute-force facet-based polytope identification (BFPI). BFPI looks for a polytope enclosing the data points by maximizing the number of points on each facet of that polytope. It relies on solving an optimization problem in the dual space. We provide an identifiability theorem for BFPI under the FBC (Theorem 4.4).
3. In Section 5, we present a greedy variant for BFPI, namely GFPI, better suited for solving practical problems. GFPI extracts the facets of $\text{conv}(W)$ containing the largest number of data points sequentially by solving mixed integer programs (MIPs). We explain how GFPI is able to handle noise, rank deficient W's, and outliers. We also provide an identifiability theorem for GFPI under the FBC (Theorem 5.5).
4. In Section 6, we show on numerous numerical experiments that GFPI outperforms the current state-of-the-art SSMF algorithms. GFPI recovers the ground truth factor W_t in much more difficult scenarios, while being less sensitive to noise and outliers.

2. Related Works: SSMF algorithms and identifiability.

Among the current approaches with identifiability guarantees for SSMF, the two main ones are arguably separable NMF [4, 5], and simplex volume minimization [36].

Separability. Separable NMF (SNMF) relies on the separability assumption. It requires that each column of W is present as a column of X, that is, that there exists an index set $\mathcal{K}$ such that $W = X(:, \mathcal{K})$. Equivalently, if separability holds, H contains the identity as a submatrix. Separability is referred to as the pure-pixel assumption in HU [8], and to the anchor word assumption in topic modeling [4]. Separability allows for efficient algorithms (that is, running in polynomial time) that are robust in the presence of noise; see [22] and the references therein. An instrumental algorithm to tackle separable NMF is the successive projection algorithm (SPA) introduced in [2], and proved to be robust to noise in [25]. However, separability is a rather strong condition and might not hold in many applications.

Minimum Volume, and Sufficiently Scattered Condition. To overcome this limitation, the Minimum-Volume (Min-Vol) framework was proposed which does not rely on the existence of the columns of W in the data set. Min-Vol looks for a simplex that encloses the data points

[2] A facet of a d-dimensional polytope is a $(d-1)$-dimensional face of that polytope. For example, in two dimensions, a polytope is a polygon and its facets are the segments.

and simultaneously has the smallest possible volume. It can be formulated as follows [18, 33]
(Min-Vol)

$$\min_{W\in\mathbb{R}^{m\times r}, H\in\mathbb{R}^{r\times n}} \det(W^\top W) \quad \text{such that} \quad X = WH \text{ and } H(:,j) \in \Delta^r \text{ for all } j.$$

When the separability assumption is violated, Min-Vol is significantly superior to SNMF. Identifiability of Min-Vol requires H to satisfy the sufficiently scattered condition (SSC), while $\text{rank}(W) = r$. For a matrix $H \in \mathbb{R}_+^{r\times n}$ to satisfy the SSC, the columns of H must be sufficiently scattered in Δ^r in order for their conical hull $\text{cone}(H) = \{y \mid y = Hx, x \geq 0\}$ to contain the second-order cone $\mathcal{C} = \{x \in \mathbb{R}_+^r \big| e^\top x \geq \sqrt{r-1}||x||_2\}$. The SSC is a much more relaxed condition than separability, see Figure 1 for an illustration. We refer the reader to [18, 13, 14] for discussions on the SSC and the identifiability of SSMF.

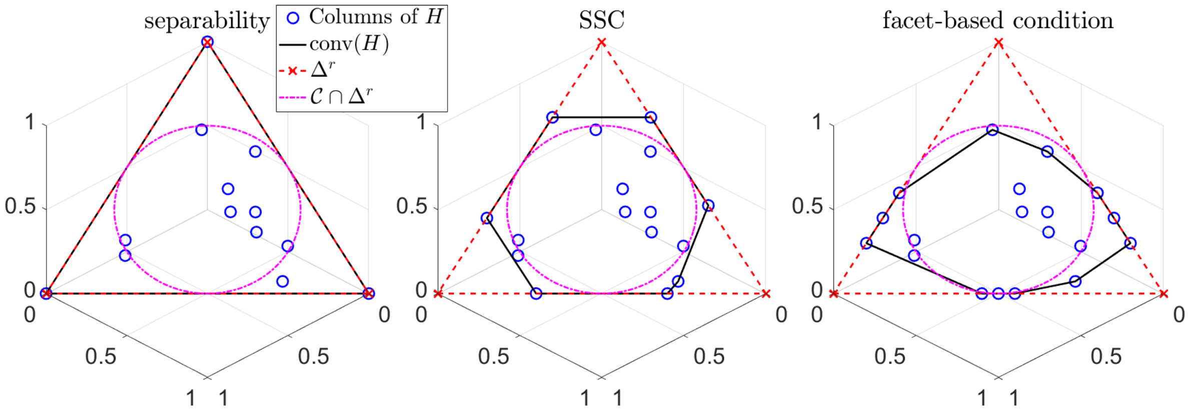


Figure 1. *Comparison of separability (left), SSC (middle), and our facet-based condition (right) for the matrix H whose columns lie on the unit simplex. On the left, separable NMF, as well as Min-Vol and FPI, will be able to uniquely identify W. On the middle, separable NMF fails while Min-Vol will uniquely identify W. Our approach may fail since the data points are also enclosed in another triangle containing six data points on its segments (there are only $r-1=2$ columns of H on each facet of Δ^r). On the right, Min-Vol fails while FPI will be able to uniquely identify W. The reason Min-Vol fails is because the triangle with minimum volume containing the data points does not coincide with Δ^r. However, the only triangle with three data points on each segment and containing all data points is Δ^r, which explains why FPI works.*

However, Min-Vol is a difficult optimization problem and, as far as we know, most methods are based on standard non-linear optimization schemes (such as projected gradient methods) come with no global optimality guarantees. Hence although Min-Vol allows for identifiability, it is still an open problem to provide an algorithm that solves the problem up to global optimality, in polynomial time; see the discussion in [14]. There exist non-ploynomial time algorithms for Min-Vol; see the next paragraph. Min-Vol has three main weaknesses:

1. It requires W to be full column rank. For example, in three dimensions, it can only identify three vertices.
2. It does not take advantage of the fact that, in many applications, most data points are usually located on the facets of the convex hull of the columns of W because H is sparse. Minimum-volume NMF only uses the columns of X that are not contained in the convex hull of the other columns, that is, the vertices of conv(X). We believe this is a crucial information to take into account, and will lead to more robust approaches:

we not only want to be able to reconstruct each data point, but also that as many points as possible are located on the facets of conv(W).

3. The SSC, although much milder than separability, is still rather strong. It might not be satisfied in highly mixed scenarios; for example when a column of W is not present in a sufficiently large proportion in sufficiently many pixels; see Figure 1 (right).

In Section 4, we will provide a new weak condition for identifiability, namely the FBC. In a nutshell, the FBC only requires to have r data points on each facet of conv(W). (Note that the SSC implies that there are at least $r-1$ data points on each of these facets.) Figure 1 illustrates the different identifiability conditions on the matrix H for $r=3$.

Improving algorithmic designs for SNMF and Min-Vol is usually the main concern of the majority of recent studies; see for example [37, 3, 17, 21, 30]. In this paper, we take another direction, and consider new identifiability conditions, along with provably correct algorithms.

Algorithms based on facet identification. As mentioned before, our model and algorithm that will be presented in Section 4 is based on the identification of the facets of conv(W). There are few representative works that are based on similar ideas.

Ge and Zou [20] introduced the concept of subset-separability which relaxes the separability condition. A factorization $X = WH$ is subset-separable if each column of W is the unique intersection point of a subset of filled facets. A facet is filled if there is at least one point in the interior of the convex hull of the columns in W corresponding to that facet or if the facet is exactly a vertex of W. This algorithm is based on finding all facets by enumerating through all columns of X. The facets are identified using the following fact: each point can be expressed as a convex combination of other points lying on the same facet. This algorithm requires the data points which are not on facets to be in general positions, so that these points cannot be identified as a filled facet. The intuition behind our approach is related to these ideas. However our proposed algorithm will be completely different and our assumptions will be weaker: we do not require the facets to be filled, and do not put a general position condition on the points within the polytope conv(W).

Lin et al. [32] proposed an algorithm that looks for the simplex enclosing the data points by determining the r associated facets, and then calculating the vertices of that simplex (that is, the columns of W) by finding the intersection of the facets. Their approach is referred to as Hyperplane-based Craig-simplex-identication (HyperCSI). The algorithm for identifying the r facets relies on SPA [2]. First, an initial estimate of the facets is computed using the r points extracted by SPA. The orientational difference between the ground-truth facet and the estimated facet is reduced by finding *active* samples that are close to the estimated facets. It was proven that in the noiseless setting, and as the number of columns of X goes to infinity, that is, $n \to \infty$, the simplex identified by HyperCSI is exactly the minimum-volume simplex.

In [34], Lin et al. proposed a different geometric approach for SSMF that is based on fitting a maximum-volume inscribed ellipsoid (MVIE) in conv(X). They show that, under the SSC, the MVIE touches every facet of conv(W) which allows it to recover them, and then W. However, computing the MVIE requires to first compute all facets of conv(W), which is NP-hard in general (the number of facets can be exponential in the number of columns of W). The second step uses semidefinite programming to compute the MVIE. As opposed to most algorithms for Min-Vol, MVIE is guaranteed to recover W in the noiseless case. However, the limitations of Min-Vol still hold here. Moreover, MVIE relies on facet enumeration which is sensitive to noise and outliers; see Section 6 for numerical experiments. This approach was recently improved by using a first-order method to solve the semidefinite program, and a different post-processing of the MVIE solution to recover W [31].

In [11], authors provide identifiability results when the input matrix H is sufficiently sparse. This result also applies to SSMF: it has a unique solution if on each subspace spanned by all but one column of W, there are $\lfloor \frac{r(r-2)}{r-k} \rfloor + 1$ data points with spark r (that is, any subset of $r-1$ columns is linearly independent). However, this is a theoretical result, with no algorithm to tackle the problem. Moreover, this result does not take nonnegativity into account, and requires much more points on each facet than our facet-based condition.

Summary. Algorithms for SSMF based on the identification of the facets of $\text{conv}(W)$ have not been very successful in practice because they are either theoretically oriented, or they rely on strong conditions and are sensitive to noise. Table 1 gives the conditions under which SSMF algorithms recover the ground truth factor W, in the noiseless case.

Table 1

Indentifiability conditions for different SSMF algorithms in the exact case. We denote $d = \text{rank}(X) \leq r$.

	# Points per facets	**separability**	**SSC**	$d = r$	$n \to \infty$
Separable NMF [2]	$d-1$	✓	✓	✓	-
Min-Vol [36]	$d-1$	-	✓	✓	-
MVIE [34]	$d-1$	-	✓	✓	-
HyperCSI [32]	$d-1$	-	✓	✓	✓
BFPI and GFPI	d	-	-	-	-

It highlights five conditions: number of points per facet of $\text{conv}(W)$ (this is essentially a sparsity condition on H), separability, SSC, full column rank of W, and whether the number of samples needs to go to infinity. Our proposed algorithms, BFPI and GFPI, require $d = \text{rank}(X)$ points per facet, which is only one additional data point on each facet compared to the other algorithms that require additional strong conditions such as the SSC or $\text{rank}(W) = r$. Hence BFPI and GFPI will not always be stronger than Min-Vol (see Figure 1), but they will be in most practical cases.

3. Identifiability of SSMF under the faced-based conditions.

Let us state the FBC.

Assumption 3.1 (Facet-based conditions (FBC)). Let $X \in \mathbb{R}_+^{m \times n}$ with $d = \text{rank}(X)$, and let $W \in \mathbb{R}^{m \times r}$ and $H \in \mathbb{R}_+^{r \times n}$ be such that $X = WH$ where

a. No column of W is contained in the convex hull of the other columns of W, that is, $\text{conv}(W)$ is a polytope with r vertices given by the columns of W.

b. The columns of H belong to the unit simplex, that is, $H(:,j) \in \Delta^r$ for $j = 1, 2, \ldots, n$.

c. Each facet of $\text{conv}(W)$ contains at least $s \geq d$ distinct columns of X and, among them, at least $d-1$ generate that facet (that is, the dimension of the convex hull of these s columns is $d-2$).

d. There are strictly less than s distinct columns of X on every facet of $\text{conv}(X)$ which is not a facet of $\text{conv}(W)$.

Let us comment on these assumptions.

- Assumption 3.1.a is necessary for any identifiable SSMF model since a column of W

cannot be identified if it is located in the convex hull of the other columns (it could be discarded to have a decomposition with $r-1$ factors).

Since $X = WH$, $d = \text{rank}(X) \leq \text{rank}(W) \leq r$. However as opposed to most previous works, we do not assume $d = r$ so that $\text{conv}(W)$ may contain more vertices than its dimension plus one; for example, it could be a quadrilateral in the plane as in Figure 3.

- Assumption 3.1.b allows for WH to be a SSMF. For NMF, that is, when $X = WH$ with $W \geq 0$ and $H \geq 0$, Assumption 3.1.b can be assumed without loss of generality by using a simple scaling of the columns of X and W; see the introduction.
- The key assumption is Assumption 2.c. It implies a certain degree of sparsity of the columns of H: a column of X is on a facet of $\text{conv}(W)$ if the corresponding column of H has at least one zero entry. Hence Assumption 2.c implies that each row of H has d zero entries, and this condition is easy to check.
- Assumption 3.1.d will allow us to make the decomposition unique. For example, assume the data points are located on the boundary of a hexagon in two dimensions with $r = 3$; see Figure 2 for an illustration. There are many possible triangles that contain these points, and SSMF is not unique. (Min-Vol) picks the unique triangle with the smallest volume, while SSMF under the FBC picks the unique triangle having three points on each segment.

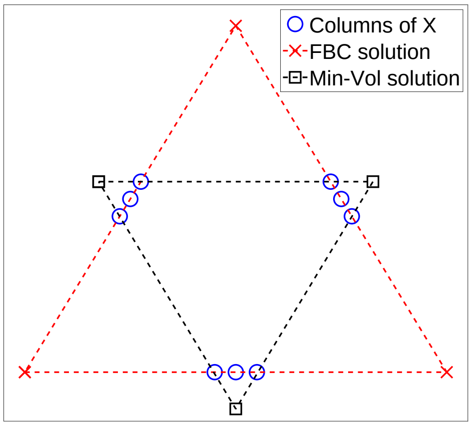


Figure 2. *Illustration of the non-uniqueness of SSMF. SSMF under the FBC achieves uniqueness based on Assumption 1.d, and selects the triangle whose vertices are the red crosses, with three points on each segment. Min-Vol selects the triangle whose vertices are the black squares, which has the smallest volume, but only two points on each segment.*

Under Assumption 3.1.d, data points can be on the boundary of $\text{conv}(X)$ as long as the number of such points on the same facet does not exceed the number of points on any of the facets of $\text{conv}(W)$. We believe that this assumption will be met in most practical situations.

Assumption 3.1.d is not easy to check as it requires to compute all facets of $\text{conv}(X)$, and there could be exponentially many. Note however that the SSC is NP-hard to check [27].

Compared to the assumption required for Min-Vol, our assumptions require one additional data points on each facet but does not require these data points to be well-spread on that facet. Moreover, we do not require X to be of rank r. Note however that the well-spreadness of data points on a facet will influence the robustness to noise of our model; see Section 6.

Remark 3.2 (Separability vs. the FBC). As opposed to the SSC, Assumption 3.1 is not a generalization of separability because a separable matrix might not satisfy Assumption 3.1.c. However, Assumption 3.1.c could be relaxed as follows: either a facet of $\text{conv}(W)$ satisfies Assumption 3.1.c or its vertices are columns of X. In that case, our results still apply, using the same trick as in [20, Algorithm 5]. We stick in this paper to Assumption 1.c for the simplicity of the presentation and because, in practice, it is not likely for a facet to contain all its vertices while not containing any point in its interior.

Before proving that the factor W in SSMF is identifiable under the FBC (Assumption 3.1), let us show the following lemma.

Lemma 3.3. *Let $X = WH$ satisfy Assumption 3.1. Then every facet of $\text{conv}(W)$ is a facet of $\text{conv}(X)$.*

Proof. Assumptions 3.1.b implies $\text{conv}(X) \subseteq \text{conv}(W)$, while each facet of $\text{conv}(W)$ contains at least d columns of X whose convex hull has dimension $d-2$ (Assumptions 3.1.c). This implies that every facet of $\text{conv}(W)$ is a facet of $\text{conv}(X)$. ∎

The proof of Lemma 3.3 leads to an interesting observation: for SSMF to be identifiable, one needs to have at least $d-1$ data points on each facet of $\text{conv}(W)$, otherwise it cannot be a facet of $\text{conv}(X)$ and hence cannot be identified. In fact, one can check that both separability and the SSC imply this condition. The FBC only requires one additional data point on each of these facets.

Theorem 3.4 (Uniqueness of W in SSMF under the FBC). *Let $X = WH$ satisfying the FBC (Assumption 3.1). For any other factorization $X = \hat{W}\hat{H}$ satisfying the FBC, $\hat{W} = W\Pi$ where $\Pi \in \{0,1\}^{r\times r}$ is a permutation matrix.*

Proof. Note that the FBC depends on the parameter $s \geq d$. Assume there exists two factorizations $X = WH$ and $X = \hat{W}\hat{H}$ satisfying the FBC (Assumption 3.1), where the parameter $s = s_W$ for WH, and $s = s_{\hat{W}}$ for $\hat{W}\hat{H}$. Assume without loss of generality that $s_W \leq s_{\hat{W}}$. By definition, the columns of W and $\hat{W}$ are the intersections of the facets of $\text{conv}(W)$ and $\text{conv}(\hat{W})$, respectively. For W and $\hat{W}$ to have at least one column that do not coincide (up to permutation), there is at least one facet of $\text{conv}(W)$ that is different from one facet of $\text{conv}(\hat{W})$. Let $\hat{\mathcal{F}}$ be a facet of $\text{conv}(\hat{W})$ that is not a facet of $\text{conv}(W)$. By Lemma 3.3, $\hat{\mathcal{F}}$ is a facet of $\text{conv}(X)$. This is in contradiction with Assumption 3.1.d for (W, H): $\hat{\mathcal{F}}$ is a facet of $\text{conv}(X)$ but not a facet of $\text{conv}(W)$ while it contains $s_{\hat{W}} \geq s_W$ distinct data points. ∎

4. Brute-force facet-based polytope identification (BFPI). In this section, we describe our first proposed algorithm, namely BFPI; see Algorithm 4.1. The high-level geometric insight of the proposed FPI algorithm is to identify the facets of $\text{conv}(W)$, given the data points. Although we will not implement BFPI, we believe the high level ideas within BFPI are key, and may be an important starting point for future algorithmic design, which is the reason why we present it here. Moreover, BFPI is provably correct and is supported by identifiability guarantees under the assumptions of the FBC; see Theorem 4.4.

Algorithm 4.1 Brute-force facet-based polytope identification (BFPI) for SSNMF

Input: Data matrix $X \in \mathbb{R}^{m \times n}$ satisfying Assumption 3.1, and parameter s.
Output: The basis matrix W.

% Step 1. Preprocessing
1: Remove the zero columns of X, and remove duplicated data points.
2: Reduce the dimension of the columns of X to a $(d-1)$-dimensional space, by constructing the matrix $\tilde{X} \in \mathbb{R}^{(d-1)\times n}$ as follows. Given the compact SVD of $X - \bar{X} = U\Sigma V^\top$ where $U \in \mathbb{R}^{m\times(d-1)}$, $\Sigma \in \mathbb{R}^{(d-1)\times(d-1)}$ and $V \in \mathbb{R}^{n\times(d-1)}$, we take

$$\tilde{X} \;=\; U^\top (X - \bar{X}) = \Sigma V^\top.$$

Let us denote $\tilde{W} = U^\top \left(W - [\bar{x} \dots \bar{x}]\right)$, so that $\tilde{X} = \tilde{W}H$.

% Step 2. Compute all vertices of $\operatorname{conv}(X)^*$
3: Compute all vertices $\{\theta_i\}_{i=1}^{v}$ of $\operatorname{conv}(X)^* = \{\theta \mid \tilde{X}^\top \theta \leq e\} \subseteq \mathbb{R}^{d-1}$.

% Step 3. Identify the vertices of $\operatorname{conv}(W)^*$
4: Identify the vertices corresponding to a facet in the primal that contain more than s points

$$J = \left\{ i \;\middle|\; \left|\{j \mid \tilde{X}(:,j)^\top \theta_i = 1\}\right| \geq s, 1 \leq i \leq v \right\}.$$

The convex hull of $\{\theta_i\}_{i\in J}$ is the dual of the convex hull of $\tilde{W}$.

% Step 4. Recover $\tilde{W}$ *from the vertices of* $\operatorname{conv}(\tilde{W})^*$
5: Recover $\tilde{W}$ by intersecting the facets $\{x \mid x^\top \theta_i \leq 1\}$ for $i \in J$.

% Step 5. Postprocess $\tilde{W}$ *to recover* W
6: Project $\tilde{W} \in \mathbb{R}^{(d-1)\times r}$ back to the original m-dimensional space: $W = U\tilde{W} + [\bar{x} \dots \bar{x}]$.

Preliminaries. Let $d = \operatorname{rank}(W)$. The facets of the $(d-1)$-dimensional polytope $\operatorname{conv}(W)$ are the polytopes of dimension $d-2$ obtained as the intersection of $\operatorname{conv}(W)$ with a hyperplane. For a set $\mathcal{A}$ containing the origin in its interior, its dual is $\mathcal{A}^* \;=\; \left\{y \mid x^\top y \leq 1 \text{ for all } x \in \mathcal{A}\right\}$. If $\mathcal{A}$ is a polytope, then $\mathcal{A}^*$ is also a polytope whose facets correspond to the vertices of $\mathcal{A}$, and vice versa. Moreover, it is easy to prove that if $\mathcal{A} \subseteq \mathcal{B}$, then $\mathcal{B}^* \subseteq \mathcal{A}^*$. We refer the reader to [44] for more information on polytopes. In order to recover the facets of $\operatorname{conv}(W)$, the dual space will be considered such that the problem of searching for the facets of a polytope is replaced by the equivalent problem of finding the vertices of a polytope in the dual space.

Preprocessing. Before doing so, the first step of FPI is to make sure the origin belongs to $\operatorname{conv}(W)$ by removing $\bar{x} = \frac{1}{n}\sum_{j=1}^{n} X(:,j)$ from all data points. This does not change the structure of the SSMF problem:

$$X(:,j) - \bar{x} = WH(:,j) - \bar{x} = (W - \bar{x}e^\top)H(:,j),$$

since $e^\top H(:,j) = 1$ because $H(:,j) \in \Delta^r$ for all j. To simplify the notation, let us denote $\bar{X} = \bar{x}e^\top$. Then, to have a full-dimensional problem, that is, to have the dimension of $\operatorname{conv}(X)$ coincide with the dimension of the ambient space, we project $X - \bar{X}$ onto its $(d-1)$-dimensional column space. In fact, since $0 \in \operatorname{conv}\left(X - \bar{X}\right)$, the rank of $X - \bar{X}$ is equal to $d-1$, and this second preprocessing step amounts to premultiplying $X - \bar{X}$ by a $(d-1)$-by-m matrix obtained via the truncated SVD of $X - \bar{X}$; see Algorithm 4.1. This does not change the structure of the SSMF problem either, it simply premultiplies X and W by a matrix of rank $d-1$. This is a standard preprocesing step in the SSMF literature; see for example [35].

Dual approach. Let us denote the dual of conv(X) as

$$\text{conv}(X)^* = \left\{\theta \mid x^\top\theta \le 1 \text{ for all } x \in \text{conv}(X)\right\} = \left\{\theta \mid X^\top\theta \le e\right\}.$$

Since conv$(X) \subseteq$ conv(W), the dual of conv(W) is contained in conv$(X)^*$.

Example 4.1. Let the columns of W be the vertices of the square $[-1,1] \times [-1,-1]$, while

$$X = \begin{pmatrix} -1 & -1 & -1 & -0.8 & -0.65 & -0.5 & -0.8 & -0.65 & -0.5 & 1 & 1 & 1 \\ 0.8 & 0.65 & 0.5 & 1 & 1 & 1 & -1 & -1 & -1 & -0.8 & -0.65 & -0.5 \end{pmatrix},$$

see Figure 3 for an illustration. The polygon conv(X) has 8 segments: 4 containing 3 data points, and 4 containing 2 data points. In the dual space, 4 of the vertices of conv$(X)^*$ correspond to the 4 vertices of conv$(W)^*$, that is, to the four segments of conv(W), while the other 4 correspond to the other 4 segments of conv(X).

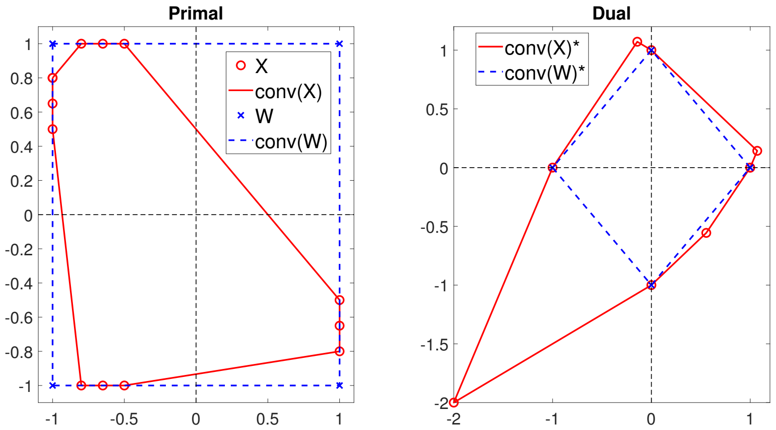


Figure 3. *Illustration of the concept of duality to compute SSMF. On the left, this is the primal space where* conv$(X) \subseteq$ conv(W)*. On the right, this is the dual representation where* conv$(W)^* \subseteq$ conv$(X)^*$*. The circles are the vertices of* conv$(X)^*$ *corresponding to the segments of* conv(X) *in the primal. The crosses are the vertices of* conv$(W)^*$ *corresponding to the segments of* conv(W) *in the primal.*

Our goal is to find the vertices of conv$(X)^*$ that correspond to the vertices of conv$(W)^*$, that is, the facets of conv(W). Under Assumption 3.1.c, there are at least d columns of X on each facet of conv(W) whose convex hull has dimension $d-2$; on Figure 3, there are three points on each segment of conv(W). This implies that a subset of the vertices of conv$(X)^*$ contains the vertices of conv$(W)^*$, as shown in the following lemma.

Lemma 4.2. *Let $X = WH$ satisfy Assumption 3.1, and assume X has been preprocessed as described in Algorithm 4.1 so that* $0 \in$ conv(X) *and $X \in \mathbb{R}^{(d-1)\times n}$ where* rank$(X) = d-1$*. Then the set of vertices of* conv$(X)^*$ *contain all the vertices of* conv$(W)^*$*.*

Proof. This follows from Lemma 3.3 and duality. ■

Once the vertices of conv$(X)^*$ are identified, we recover the vertices of conv$(W)^*$ that correspond to the facets of conv(W) containing the largest number of data points. More precisely, under Assumption 3.1, we have the following lemma.

Lemma 4.3. *Let $X = WH$ satisfy Assumption 3.1, and assume X has been preprocessed as described in Algorithm 4.1 so that $0 \in \mathrm{conv}(X)$, $X \in \mathbb{R}^{(d-1)\times n}$ where $\mathrm{rank}(X) = d-1$, and X does not have duplicated columns. Then the set $\{x \in \mathrm{conv}(W) \mid \theta^\top x = 1\}$ for $\theta \in \mathbb{R}^{d-1}$ is a facet of $\mathrm{conv}(W)$ if and only if*

$$\text{(4.1)} \qquad \theta \text{ is a vertex of } \mathrm{conv}(X)^* = \left\{\theta \mid X^\top \theta \leq e\right\} \quad \text{and} \quad \left|\{j \mid X(:,j)^\top \theta = 1\}\right| \geq s,$$

where $|\mathcal{A}|$ denotes the cardinality of the set $\mathcal{A}$.

Proof. Let $\{x \in \mathrm{conv}(W) \mid \theta^\top x = 1\}$ be a facet of $\mathrm{conv}(W)$. By Lemma 3.3, θ must belong to $\mathrm{conv}(X)^*$, while, by Assumption 3.1.c, facets of $\mathrm{conv}(W)$ contain more than $s \geq d$ columns of X.

Let θ satisfy (4.1) so that the set $\mathcal{F} = \{x \in \mathrm{conv}(W) \mid \theta^\top x = 1\}$ contains s columns of X. Since θ is a vertex of $\mathrm{conv}(X)^*$, the set $\mathcal{F}$ corresponds, by duality, to a facet of $\mathrm{conv}(X)$. By Assumption 1.c, the facets containing at least s points must correspond to facets of $\mathrm{conv}(W)$. ■

Finally, W is recovered by intersecting the facets of $\mathrm{conv}(X)$ containing more than s data points. The proposed brute-force algorithm is presented in Algorithm 4.1. The main step of Algorithm 4.1 is a vertex enumeration problem in the dual space.

Identifiability. Let us prove that, if $X = WH$ satisfies Assumption 3.1, then Algorithm 4.1 recovers W, up to permutation of its columns.

Theorem 4.4 (Recovery of W by Algorithm 4.1). *Let $X = WH$ satisfy Assumption 3.1. Then Algorithm 4.1 recovers the columns of W (up to permutation).*

Proof. First, as already noted above, the prepossessing step does not change the geometry of the problem, that is, if $X = WH$ satisfies Assumption 3.1, then $\tilde{X} = \tilde{W}H$ also satisfies Assumption 3.1. Hence let us assume w.l.o.g. that $0 \in \mathrm{conv}(X)$ and $X \in \mathbb{R}^{(d-1)\times n}$ where $\mathrm{rank}(X) = d-1$. The rest of the proof follows from Lemmas 3.3 and 4.3. By Lemma 3.3, the vertices of $\mathrm{conv}(X)^*$ computed in step 4 of Algorithm 4.1 correspond to facets of $\mathrm{conv}(X)$. By Lemma 4.3, only the facets of $\mathrm{conv}(X)$ corresponding to facets of $\mathrm{conv}(W)$ containing at least s columns of X. ■

Computational cost. Algorithm 4.1 may run in the worst-case in exponential time. The set $\mathrm{conv}(X)^*$ is an $(d-1)$-dimensional polytope defined by n inequalities and can have exponentially many vertices, namely $O\left(\binom{n}{d-1}\right)$.

Although we could adapt BFPI to handle noisy input matrices, we will develop in the next section a more practical algorithm that does not require to identify all vertices of $\mathrm{conv}(X)^*$, and that can handle noise and outliers. However, we believe BFPI is important, and could be the starting point for other practical SSMF algorithms.

5. Greedy FPI (GFPI).

The brute-force approach presented in the previous section is provably correct but may require exponentially many operations. Note that the same observation holds for (Min-Vol): as far as we know, the algorithms that provably solve (Min-Vol) up to global optimally require to compute all facets of $\mathrm{conv}(X)$; see Section 2. In this section, we propose a practical sequential algorithm, dubbed Greedy FPI (GFPI), by leveraging highly efficient MIP solvers (in particular their ability to quickly find high quality solutions). Although it is still computationally heavy to solve (that is, we cannot prove it runs in polynomial time), it allows to solve large problems; see Section 6.

GFPI sequentially searches for the facets of $\text{conv}(X)$ containing the largest number of points. This section is organized as follows. The optimization model used to identify a facet, even in the presence of noise and outliers, is described in Section 5.1. Once a facet is identified, the same model can be used to extract the next facet, by removing the previously identified facets from the search space (Section 5.2). To make sure the intersection of the r extracted facets corresponds to a bounded polytope, we add a constraint when extracting the last facet (Section 5.3). The way the matrix W is estimated from the extracted facets is described in Section 5.5. Finally, in Section 5.6, we prove the identifiability of GFPI under the FBC, and discuss its computational cost and the choice of its parameters.

5.1. Identifying a facet, in the presence of noise and outliers. As for GFPI, the data points are first centered and projected into a $(d-1)$-dimensional subspace to obtain $\tilde{X} \in \mathbb{R}^{(d-1)\times n}$ such that $0 \in \text{conv}\left(\tilde{X}\right)$ and $\text{rank}\left(\tilde{X}\right) = d-1$. Since we want GFPI to handle noisy data, we cannot use the metric of the number of points on a facet of $\text{conv}\left(\tilde{X}\right)$ to know whether it is also a facet of $\text{conv}\left(\tilde{W}\right)$, because points will not be exactly located on the facets of $\text{conv}(X)$. Given a parameter γ that depends on the noise level, we propose to solve

$$\max_{\theta\in\mathbb{R}^{d-1}} \sum_{j=1}^{n} \mathbf{I}\left(\tilde{X}(:,j)^\top\theta \geq 1-\gamma\right) \quad \text{such that} \quad \tilde{X}^\top\theta \leq (1+\gamma)e, \tag{5.1}$$

where $\mathbf{I}(.)$ is the indicator function which is equal to 1 if the input condition is met, and to 0 otherwise. The variable θ encodes the facet $\{x \in \text{conv}\left(\tilde{X}\right) \mid x^\top\theta = 1\}$. The optimal solution of (5.1) corresponds to a facet containing the largest number of data points within a safety gap defined by γ. In the noiseless case, taking $\gamma = 0$ and solving (5.1) provides a facet of $\text{conv}(X)$ containing the largest number of columns of X, and hence it will correspond to a facet of $\text{conv}(W)$, under Assumption 3.1; see Lemma 4.3.

To solve (5.1), we use a MIP. We introduce a binary variable $y_i \in \{0,1\}$ $(1 \leq i \leq n)$ which is equal to 0 if $\mathbf{I}(\tilde{X}(:,i)^\top\theta \geq 1-\gamma) = 1$, and to 1 otherwise[3], and solve

$$\min_{\theta\in\mathbb{R}^{d-1}, y\in\{0,1\}^n} \sum_{j=1}^{n} y_j \quad \text{such that } 1-\gamma-Ay_j \leq \tilde{X}(:,j)^\top\theta \leq 1+\gamma \text{ for } 1 \leq j \leq n.$$

The parameter A is a sufficiently large scalar based on the BIG-M approach often used to model indicator functions; see Remark 5.1. If the condition $\tilde{X}(:,j)^\top\theta \geq 1-\gamma$ is satisfied, the value of y_j can be either 0 or 1. Since the MIP minimizes y_j, y_j will be set to 0. If it is not satisfied, that is, $\tilde{X}(:,j)^\top\theta < 1-\gamma$, then the value of y has to be equal to 1. Note that $y_j = 0$ means that the corresponding data point is located close to the sought facet.

We have observed numerically that using the same safety gap for the n constraints $\tilde{X}^\top\theta \leq (1+\gamma)e$ does not give enough degrees of freedom to the formulation, and, in difficult scenarios, fails to return good solutions. In particular, it is unable to deal with outliers that might be arbitrarily far away from the sought polytope of which $\{x \mid \theta^\top x \leq 1\}$ is a facet. Hence we

[3]We made this (arbitrary) choice to obtain a minimization problem, which is more standard.

introduce the variable $\delta \in \mathbb{R}^n_+$ that accounts for the distance of the data points from the polytope; in particular, $\delta_j = 0$ if $\theta^\top \tilde{X}(:,j) \leq 1$. We propose the following MIP

$$\min_{\theta,\delta\geq 0,y\in\{0,1\}^n} \sum_{j=1}^{n} y_j + \lambda \sum_{j=1}^{n} \delta_j \text{ such that } 1-\gamma - Ay_j \leq \tilde{X}(:,j)^\top \theta \leq 1+\delta_j \text{ for } 1 \leq j \leq n,$$

$$\delta_j \leq Ay_i + \gamma \text{ for } 1 \leq j \leq n. \tag{5.2}$$

The parameter λ controls how much the points are allowed to be far away from the polytope. The constraint $\delta_j \leq Ay_i + \gamma$ forces the binary variable y_j to get the value of 0 only when $\left|\tilde{X}(:,j)^\top\theta - 1\right| \leq \gamma$, so that the data point is in fact close to the facet, up to the safety gap γ. The entries of δ larger than γ will correspond to outliers, that is, points that are outside and far away from the sought polytope.

Remark 5.1 (Value of A). The BIG-M formulation is frequently used as a modeling trick for problems with disjunctive or indicator constraints [7]. Choosing a good value for A is a difficult problem in the MIP literature. A good choice for the parameter A depends on the data. A very large value for A leads to weak relaxations, while a very small value removes feasible solutions. We have set A to 10 in all the experiments in the absence of outliers and did not notice sensitivity to this value. For the experiments with outliers, we used $A = 100$; this makes sense as outliers are further away from $\operatorname{conv}(W)$.

5.2. Cutting previous facets from the solution space. Solving (5.2) allows to approximate one facet of $\operatorname{conv}\left(\tilde{W}\right)$. In order to extract other facets sequentially, we need to eliminate the previously found facets from the feasible solutions of (5.2). To do so, we select one point in each of the previously identified facets such that it *only* belongs to the corresponding facet, that is, it needs to be in the relative interior of that facet. This point is chosen as the average of the data points associated to that facet. We will denote $M^{(t)} \in \mathbb{R}^{(d-1)\times t}$ the matrix whose columns correspond to these points after t facets have been identified. At the next step, that is, at the $(t+1)$th step, we restrict the search space of (5.2) by adding the following constraints making sure that these selected points do not lie on the current sought facet:

$$\theta^\top M^{(t)}(:,i) \ \leq 1-\gamma-\eta \quad \text{for } i = 1,\ldots,t,$$

where $\eta \in \mathbb{R}_+$ is a margin parameter which controls how far the next facet should be from the previously selected facets. The larger η is, the further the facets will be from each other. Figure 4 illustrates this procedure after one facet has been identified (corresponding to θ_1 on the figure), in the primal and dual spaces simultaneously. As the margin parameter η increases, more and more feasible solutions are cut from the dual $\operatorname{conv}(X)^*$. However, for all margin values, namely $\{0.1, 0.5, 0.8\}$, the two other vertices of $\operatorname{conv}(W)^*$ are not cut. In general, if the margin value η is set too high, there will be no feasible solution to the optimization problem and, if it is set too low, the algorithm might find a facet too close to the previously identified facets. However, both cases can be prevented. If the optimizing algorithm does not find any feasible solution, the margin can be reduced. If the identified facet is not sufficiently different from the other ones, it can be increased. However, as shown in Section SM1.4 (supplementary material), our approach is not too sensitive to this parameter.

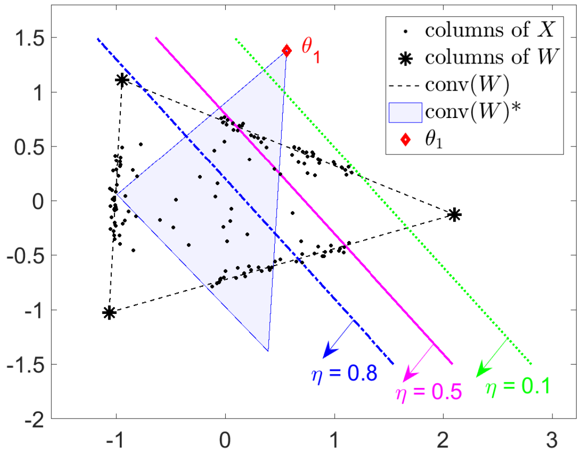


Figure 4. *Illustration of the effect of the margin parameter η on the solution space with $r = 3$.*

Remark 5.2 (Construction of $M^{(t)}$). If the data points associated to a facet are not well-spread in that facet, their average might lie near the boundary of that facet. (Note however that, by Assumption 3.1, the points on a facet generate that facet hence their average has to be in the relative interior of the facet.) In this situation, a separable NMF algorithm, such as SPA, can be used to identify $d-1$ points well spread on this facet, and then take the average of this subset of points. In this paper, we use the successive nonnegative projection algorithm (SNPA) [21] which is more robust to noise than SPA. This second strategy is useful in more difficult scenarios, and we have used it for the real-world hyperspectal images in Section 6.2.

5.3. Obtaining a polytope. We are now able to extract sequentially facets of conv(X) that approximately contain the largest number of columns of X. Let us focus on the case W is full column rank, that is, rank$(W) = r$. In difficult scenarios, for example when W is ill-conditioned, or the noise level is high, we cannot guarantee that, after having extracted r facets, we will obtain a polytope (that is, a bounded polyhedron). In order to resolve this issue, we take advantage of the following theorems.

Theorem 5.3 (Boundedness theorem [38]). *Let $\theta_1, \dots, \theta_d$ be d linearly independent vectors in $\mathbb{R}^d$. If $\theta^{d+1} = -\sum_{i=1}^{d} \mu_i \theta^i$ with $\mu > 0$, then the positive hull of these $d+1$ vectors span $\mathbb{R}^d$.*

Theorem 5.4 (Full body theorem [38]). *Given a set $\Theta = \{\theta_1, \dots, \theta_\ell\}$ in $\mathbb{R}^d$, the polyhedron $\mathcal{P} = \{x | \theta_i^\top x \leq b_i; i = 1, \dots, \ell\}$ is bounded if and only if the positive hull of Θ spans $\mathbb{R}^d$.*

To ensure that the r identified facets define a bounded polytope in $\mathbb{R}^{d-1}$, we add the following constraint to (5.2) when computing the last facet:

$$\theta = -\sum_{i=1}^{d-1} \mu_i \theta^{(i)} \quad \text{with} \quad \mu_i \geq \epsilon \text{ for } i = 1, \dots, d-1, \tag{5.3}$$

where $\theta^{(i)}$ $(1 \leq i \leq r-1)$ are the $r-1$ vectors extracted at the first $r-1$ steps of GPFI, and ϵ is a small positive constant. We used $\epsilon = 0.1$ in all numerical experiments in Section 6.

As mentioned above, this additional constraint plays an instrumental role in difficult scenarios. For example, on the real hyperspectral images from Section 6.2 that are highly contaminated with noise (and do not follow closely the model assumptions), this constraint allowed us to obtained significantly better solutions; see in particular Figure 9-(b) where one of the extracted facet does not have many points around it: its extraction was made possible because of (5.3). Moreover, we have observed that the use of (5.3) makes the identification of the last facet less sensitive to the margin parameter η as (5.3) forces the sought facet to be far from the facets already identified.

Rank-deficient case. Our sequential strategy can extract more than r facets of $\operatorname{conv}(W)$ when $\operatorname{rank}(W) < r$; for example, in Section 6.1.3, we will extract the 4 segments of a square. In practice, in the rank-deficient case, it is unclear how many facets need to be extracted. In two dimensions, the number of facets of a polygon coincides with the number of vertices. However, in higher dimensions, the number of facets and vertices cannot be deduced from one another. Hence we leave to the user to decide how many facets are extracted. A possible heuristic would be to extract facets as long as they contain sufficiently many data points, and/or as long as the corresponding polyhedron is unbounded. We leave this as a direction of further development.

5.4. Summary of the MIP model for facet identification. To summarize, GFPI will extract one facet at each iteration. At iteration t, it solves the following MIP:

$$\min_{\theta\in\mathbb{R}^{d-1},\delta\in\mathbb{R}^n_+,y\in\{0,1\}^n} \sum_{j=1}^{n} y_i + \lambda \sum_{j=1}^{n} \delta_j \tag{5.4}$$

$$\begin{aligned}
\text{such that } & \tilde{X}(:,j)^\top \theta \leq 1 + \delta_j \text{ for } 1 \leq j \leq n, && \to \text{Forming dual space}\\
& \tilde{X}(:,j)^\top \theta \geq 1 - \gamma - A y_j \text{ for } 1 \leq j \leq n, && \to \text{Counting points on the facet}\\
& \theta^\top M^{(t-1)}(:,k) \leq 1 - \gamma - \eta \text{ for } 1 \leq k \leq t-1, && \to \text{Removing previous facets}\\
& \delta_j \leq A y_j + \gamma \text{ for } 1 \leq j \leq n. && \to \text{Discarding outliers}
\end{aligned}$$

The optimal solution of (5.4) at iteration t for the variable θ will be denoted $\theta^{(t)}$, it approximates the tth facet of $\operatorname{conv}\big(\tilde{W}\big)$. When $\operatorname{rank}(W) = r$, the constraint (5.3) is added when extracting the last facet to obtain a bounded polytope; see Section 5.3.

The proposed MIP model (5.4) has been carefully designed in order to achieve state-of-the-art performances on synthetic and real-world data sets; see Section 6 for the numerical experiments. It results from a long trial-and-error procedure, and many alternative formulations have been tested. A direction of research is to further improve this MIP formulation.

5.5. Post-processing: intersection of facets. Once the facets of $\operatorname{conv}\big(\tilde{W}\big)$ are identified, that is, the vectors $\{\theta^{(t)}\}_{t=1}^{T}$ are computed sequentially using (5.4), how can we recover $\tilde{W}$ accurately, even in noisy conditions? It is possible to improve the quality of the identified facets, and hence of $\tilde{W}$, by taking advantage of the knowledge of the data points associated to them. For the identified facet corresponding to $\theta^{(t)}$ ($1 \leq t \leq T$), let

$$J^{(t)} = \left\{ j \;\middle|\; \left| \tilde{X}(:,j)^\top \theta^{(t)} - 1 \right| \leq \gamma \right\}$$

be the index set containing the points associated to it. The set $J^{(t)}$ contains the indices such that $y_j = 0$ when solving (5.4). To improve the estimate of $\theta^{(t)}$, we compute the normal vector of the affine hull containing the columns of $X(:, J^{(t)})$, which is the left singular vector corresponding to the smallest singular value of the SVD of $X(:, J^{(t)})$, after removing the average from each column (the facet is translated so that 0 belongs to it). Let us denote $\Theta \in \mathbb{R}^{d-1\times T}$ the matrix whose columns are these singular vectors so that $\Theta(:, t)$ replaces $\theta^{(t)}$. The facet t has the form $\{x \mid \Theta(:, t)^\top x = q_t\}$ for some offset q_t. Again, we compute q_t from the data by taking the average dot product between the normal vector $\Theta(:, t)$ with the data points associated to that facet, that is, we take

$$q_t = \frac{\Theta(:, t)^\top X(:, J^{(t)})e}{|J^{(t)}|} \quad \text{for} \quad t = 1, 2, \ldots, T.$$

Finally, our estimation of the polytope conv $\left(\tilde{W}\right)$ is given by $\mathcal{P} = \{x \mid \Theta^\top x \leq q\}$. Estimating $\tilde{W}$ from $\mathcal{P}$ can be done using any off-the-shelf vertex enumeration algorithm. We have used the approach in [9] whose implementation is provided in [28].

Finally, to estimate the matrix W, our estimated $\tilde{W}$ is projected back onto the original m-dimensional space, as in Algorithm 4.1.

5.6. Identifiability. Algorithm 5.1 provides the pseudo-code for GFPI. The main difference with BFPI (Algorithm 4.1) is the way the facets of $\text{conv}(\tilde{W})$ are extracted.

For well-chosen parameters, GFPI recovers the unique SSMF under the FBC.

Theorem 5.5. *Let $X = WH$ satisfy the FBC (Assumption 3.1). Let also the parameters of GFPI (Algorithm 5.1) be as follows: $\gamma = 0$, η is sufficiently small, $\lambda \to +\infty$, A is sufficiently large, T is the number of facets of $\text{conv}(W)$, and $d = \text{rank}(X)$. Then Algorithm 5.1 recovers the columns of W (up to permutation).*

Proof. The preprocessing ensures that $0 \in \text{conv}(\tilde{X})$ and $\tilde{X} \in \mathbb{R}^{(d-1)\times n}$ where $\text{rank}(\tilde{X}) = d - 1$, while the geometry of the problem remains unchanged, as in Theorem 4.4.

Let us discuss the parameters and their influence on (5.4):

- The variable δ was introduced to handle noise; see Section 5.1. Taking $\lambda \to +\infty$ implies that the optimal solution for the variable δ in (5.4) is 0, because $\delta = 0$ is part of many feasible solutions (take for example any θ such that $\tilde{X}^\top\theta \leq e$, such as $\theta = 0$ since $0 \in \text{conv}(\tilde{X})$, and $y_j = 1$ for all j). In other words, in the noiseless case, δ can be set to zero and removed from the formulation (5.4). Note that for $\delta = 0$, the first constraint of (5.4) reduces to forming the dual space, that is, $\tilde{X}^\top\theta \leq e$, while the last constraints, dealing with outliers, can be removed since $A, y, \gamma \geq 0$.
- For A sufficiently large and $\gamma = 0$, the objective of (5.4) is equivalent to the indicator function counting the points on the facet $\{x \mid \theta^\top x = 1\}$; see Section 5.1.

This means that, for the chosen parameters, (5.4) is equivalent to

$$\text{(5.5)} \qquad \max_{\theta\in\mathbb{R}^{d-1}} \sum_{j=1}^{n} \mathbf{I}\left(\tilde{X}(:, j)^\top\theta \geq 1\right) \quad \text{such that} \quad \tilde{X}^\top\theta \leq e \;\text{ and }\; {M^{(t-1)}}^\top\theta \leq (1-\eta)e.$$

Now, let us prove the result by induction.

Algorithm 5.1 Greedy FPI (GFPI)

Input: Data matrix $X \approx WH \in \mathbb{R}^{m\times n}$ satisfying Assumption 3.1 approximately, number T of facets to extract, dimension d, and the parameters $\gamma \geq 0$, $\eta > 0$, $\lambda > 0$, and $A > 0$.

Output: Recover the basis matrix $W \in \mathbb{R}^{m\times r}$ approximately.

% Step 1. Preprocessing

Use the same preprocessing as in Algorithm 4.1, to obtain $\tilde{X} = U^\top[X - \bar{X}] \in \mathbb{R}^{(d-1)\times n}$.

% Step 2. Extract the T facets of $\operatorname{conv}\left(\tilde{W}\right)$

Initialization: Set $M^{(0)} = [\,]$, and $\Theta = [\,]$.

for $t = 1, 2, \ldots, T$ **do**

Compute $\theta^{(t)}$ as the optimal solution of (5.4). If $t = T = d$, use the additional constraint (5.3) within (5.4) to obtain a bounded polytope.

Identify the data points close to the facet corresponding to $\theta^{(t)}$, that is,

$$J^{(t)} = \left\{ j \mid \left| \tilde{X}(:,j)^\top \theta^{(t)} - 1 \right| \leq \gamma \right\}.$$

Compute the average of these points as $m^{(t)} = \frac{\tilde{X}(:,J^{(t)})e}{|J^{(t)}|}$, and let $M^{(t)} = [M^{(t-1)}, m^{(t)}]$.

Provide a more reliable estimate of $\theta^{(t)}$: add as a column of Θ the left singular vector of $\tilde{X}(:,J^{(t)}) - [m^{(t)} \ldots m^{(t)}]$ corresponding to its smallest singular value.

Compute the tth entry of the offset vector, $q_t = \frac{\Theta(:,t)^\top X(:,J^{(t)})e}{|J^{(t)}|} = \Theta(:,t)^\top m^{(t)}$.

end for

% Step 3. Recover $\tilde{W}$

Compute the columns $\tilde{W}$ as the r vertices of the polytope $\{x \mid \Theta^\top x \leq q\}$.
If $T = d$, then $r = T = d$: it is equivalent to solving the linear systems $\Theta(:,\bar{k})^\top \tilde{W}(:,k) = q(\bar{k})$ for $k = 1, 2, \ldots, r$ where $\bar{k} = \{1, 2, \ldots, r\} \backslash \{k\}$.

% Step 4. Postprocess $\tilde{W}$ to recover W

Project $\tilde{W} \in \mathbb{R}^{(d-1)\times r}$ back to the original m-dimensional space: $W = U\tilde{W} + [\bar{x} \ldots \bar{x}]$.

First step. Solving (5.5) boils down to maximizing the number of data points in the set $\{x \in \operatorname{conv}(\tilde{X}) \mid \theta^\top x = 1\}$. By Lemma 4.3, this is a facet of $\operatorname{conv}(\tilde{W})$; in fact, it is a facet containing the largest number of data points.

Induction step. Assume GFPI has extracted k facets of $\operatorname{conv}(\tilde{W})$. The columns of $M^{(k)}$ are located in the relative interior of their corresponding facets. This follows from Assumption 3.1.c because data points on that facet of $\operatorname{conv}(\tilde{W})$ generate that facet. Because of the constraint ${M^{(k)}}^\top \theta \leq (1-\eta)e$, the previously extracted $\theta^{(t)}$ $(1 \leq t \leq k)$ are eliminated from the feasible set of (5.5), because $M^{(k)}(:,t)^\top \theta^{(t)} = 1$ for $1 \leq t \leq k$. Moreover, for η sufficiently small, no other vertex of $\operatorname{conv}(\tilde{W})^*$ is cut from the feasible set (see Figure 4 for an illustration). In fact, for $\eta \to 0$, only the vertices $\theta^{(t)}$ $(1 \leq t \leq k)$ are cut from $\operatorname{conv}(\tilde{X})^*$. Therefore the next step of GFPI identies a facet of $\operatorname{conv}(\tilde{X})$ not extracted yet and containing the largest possible number of points. By Assumption 3.1.c-d, this must correspond to a facet of $\operatorname{conv}(\tilde{W})$. At the last step when $t = T$ and if $T = d$, the constraint (5.3) is added to (5.5). Since $\operatorname{conv}(\tilde{W})$ is bounded, by definition, it does not prevent the model to extract the last facet of $\operatorname{conv}(\tilde{W})$. It was used as a safety constraint in difficult scenarios; see Section 5.3. ■

In Section 6.1, we will show that GFPI in fact performs perfectly in noiseless conditions under Assumption 3.1. An important direction of research is to characterize the robustness to noise of GFPI. This is also an open problem for algorithms based on Min-Vol; see Section 2.

5.7. Computational cost. Identifying each facet requires to solve the MIP (5.4). Solving MIPs is in general NP-hard and can be time consuming. In fact, the proposed model can be hard to solve up to global optimality when n and/or r become large. Moreover, we have observed that, as the noise level increases, the problem gets more challenging which increases the computational time as well. We will use IBM-CPLEX (v12.10) [12] for solving the MIP (5.4). We noticed that CPLEX is able to find the optimal solution quite fast in many cases, even though it might require a lot of time to certify global optimality. In Table 3 (Section 6.1.1), we will perform such a numerical experiment: for example, for $m = r = 6$ and $n = 190$, CPLEX finds the 6 facets in 1.44 seconds on average, while it requires 2700 seconds to provide an optimality certificate. Moreover, CPLEX is often able to find good feasible solutions quickly, and hence can be stopped early providing reasonable solutions for GFPI. In Section 6.2, we will use a time limit of 100 seconds for each facet identification on two large real data sets, and GFPI will provide solutions whose quality is similar to the state of the art. Interestingly, this observation holds even for problems with dimensions as large as 30. For example, in the noiseless case and for $d \leq 30$, CPLEX finds in most case the optimal solution for each facet in less than 100 seconds[4]. In the supplementary material SM1.3, we provide additional numerical experiments on the computational cost of GFPI. A direction of further research would be to design dedicated algorithms (including heuristics) to tackle (5.4), taking advantage of its particular structure and geometry.

Remark 5.6 (Convex relaxation of the MIP (5.4) in GFPI). The core optimization problem (5.4) in GFPI is a MIP, and the constraints and the objective function are linear. Hence a natural idea to find an approximate solution of (5.4) is to relax the binary constraints on y by $0 \leq y \leq 1$ to obtain a linear program (LP). However, our numerical experiments show that this approach leads to bad solutions and poor performance in most cases. Note that CPLEX is based on branch and bound where the solution of the relaxed LP is the first computed solution, at the root node [41].

5.8. Robustness to noise. A challenging research direction is the design of SNMF algorithms without the separability assumption and that are provably robust against noise. In fact, to the best of our knowledge, the only such algorithm available in the literature is the one from [20] which relies on strong assumptions and has not been shown yet to compete with state-of-the-art algorithms on practical problems (see Section 2). In particular, proving robustness of algorithms based on the SSC and the Min-Vol framework is a major missing piece in the literature of SSMF [14]. However, Min-Vol algorithms have been shown to work well in noisy scenarios; see for example [1] and the references therein.

Algorithms based on facet identification, such as GFPI and the ones discussed in the introduction, could be rather sensitive to noise. As least they have not been used as much as Min-Vol algorithms in practice. Intuitively, in noisy scenarios, it may be difficult to identify the facets of a polytope, while one may identify hyperplanes inside the polytope as facets. The later problem is in fact an issue for the algorithm of [20] where authors need to assume

[4]For synthetic data sets, in the noiseless case, we know the optimal solution which allows us to check whether CPLEX found it. As shown in Section SM1.3, for CPLEX to return the global optimal solution with a certificate takes more than one hour, even for small values of r and n.

that points that are not on a facet of $\text{conv}(W)$ are in general position. However, GFPI is less sensitive to inner hyperplanes as it requires all data points to be located on one side of the facet; see the first constraint of (5.4). Moreover, as we will see in Section 6.1.2 and the supplementary material SM1.1, GFPI will show encouraging robustness in identifying facets for corrupted data with moderate level of noise. Moreover, we do believe that if the parameters of GFPI are properly chosen, it is not significantly impacted by the inner hyperplanes within the polytope. Recall that the parameters λ and γ indicate how deep GFPI looks for hyperplanes within $\text{conv}(W)$. For a detailed analysis of the effect of these parameters on the performance of GFPI, we refer to the discussion and numerical experiments in the supplementary material SM1.4.

Of course, as for Min-Vol algorithms, analyzing the robustness of GFPI is an important research direction. It would require to adapt the FBC. In fact,

- The noise allowed for GFPI to approximately recover W will depend on the conditioning of $\text{conv}(W)$, as for separable NMF algorithms. For example, less noise can be added to a flat triangle than to a equilateral one. This requires to adapt Assumption 3.1.a by requiring the conditioning of $\text{conv}(W)$ to be lower bounded by a positive number.
- Data points on the facets of $\text{conv}(W)$ should be well spread on that facet. For example, if all data points on a facet are very close to one another, it will be harder to accurately estimating the corresponding facet in the presence on noise. This requires to adapt Assumption 3.1.c.
- Facets of $\text{conv}(X)$ that are not facets of $\text{conv}(W)$ cannot have too many points in their neighborhood. This requires to adapt Assumption 3.1.d.

Such a theoretical robustness analysis is highly challenging and out of the scope of this paper. We leave it as a future work.

6. Numerical Experiments. In this section, GFPI is evaluated on synthetic and real-world dat sets. All experiments are implemented in Matlab (R2019b), and run on a laptop with Intel Core i7-9750H, @2.60 GHz CPU and 16 GB RAM. We use IBM-CPLEX (v12.10) [12] for solving the MIP (5.4). The code is available from https://sites.google.com/site/nicolasgillis/code, and all experiments presented in this paper can be reproduced using this code. Note that the user can also use the Matlab MIP solver, `intlinprog`, which may be convenient.

Compared Algorithms. GFPI is compared with the following state-of-the-art algorithms:

- Successive nonnegative projection algorithm (SNPA) [21]: This is an extension of SPA which is provably more robust to noise, and can handle rank deficient matrices.
- Simplex volume minimization: We use the model

$$\min_{W,H} \|X - WH\|_F^2 + \tilde{\lambda} \operatorname{logdet}(W^\top W + \delta I_r) \quad \text{such that } H(:,j) \in \Delta^r \text{ for all } j, \tag{6.1}$$

  which has been shown to provide the best practical performances [16, 1], and use the efficient algorithm proposed in [29]. We will use different parameters for $\tilde{\lambda} = \lambda \frac{\|X - W^{(0)}H^{(0)}\|_F^2}{\operatorname{logdet}(W^{(0)\top} W^{(0)} + \delta I_r)}$ where $(W^{(0)}, H^{(0)})$ is computed by SNPA, while $\delta = 0.1$; see [29] for more details. We refer to this algorithm as min vol.
- Maximum volume inscribed ellipsoid (MVIE) [34], see Section 2.
- Hyperplane-based Craig-simplex-identication (HyperCSI) [32], see Section 2.

Quality measures. To quantify the performance of SSMF algorithms, the following metrics will be used. For the synthetic data experiments, we will use the relative distance between the ground-truth W_t and the estimated W

$$\text{ERR} = \frac{||W_t - W||_F}{||W_t||_F},$$

where the columns of W are permuted to minimize this quantity, using the Hungarian algorithm. For real hyperspectral images, we will use the average mean removed spectral angle (MRSA) between the columns of W and W_t (after a proper permutation of the columns of W). This is the most common choice in this area of research. The MRSA between two vectors $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^n$ is

$$\text{MRSA}(x, y) = \frac{100}{\pi} \cos^{-1}\left(\frac{(x - \bar{x}e)^\top (y - \bar{y}e)}{||x - \bar{x}e||_2 ||y - \bar{y}e||_2} \right),$$

where $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$. We will also use the relative reconstruction error, $\text{RE} = \frac{||X - WH||_F}{||X||_F}$.

6.1. Synthetic data sets.

In this section, we compare GFPI with the state-of-the-art approaches on synthetic data sets.

Data generation. To generate full-rank synthetic data sets $X = W_t H_t$, we follow a standard procedure; see for example [1]. Each entry of W_t is drawn uniformly at random from the interval $[0, 1]$. We discard the matrices with condition number larger than $10r$ to avoid too ill-conditioned matrices.

We generate the columns of matrix H_t by splitting them in two parts: $H_t = [H_1, H_2]$. The matrix H_1 corresponds to the points lying on facets, making sure there are enough points on each facet so that Assumption 3.1 holds. The matrix H_2 corresponds to data points randomly generated within $\text{conv}(W)$. We generate H_1 and H_2 as follows.

1. Let n_1 be the number of data points on each facet. For each sample on a facet, the corresponding $r - 1$ nonzero elements in the columns of H_1 are generated using the Dirichlet distribution with parameters equal to $\frac{1}{r-1}$.
2. Let n_2 denotes the number of samples within the simplex, possibly lying on some facets but this is not strictly enforced. The columns of H_2 are generated by the Dirichlet distribution with parameters set to $\frac{1}{r}$.

Let us define the purity parameter $p \in (0, 1]$ used to quantify how far the columns of X are from the columns of W_t. It is defined as $p(H_t) = \min_{1 \leq k \leq r} ||H_t(k, :)||_\infty$. Recall that each row of H_t corresponds to the activation of the corresponding column of W, while $H_t(:, j) \in \Delta^r$ for all j. Therefore, $p(H_t)$ indicates how much the separability assumption is violated. For $p(H_t) = 1$, X satisfies the separability assumption since each column of W_t appears in the data set. For $p(H_t) = 0$, at least one of the columns of W is not used to generate X. In order to control the purity of H_t, that is, $p(H_t)$, we use the parameter p, and resample the columns of H_1 and H_2 with entries larger than[5] p, that is, we define an upper bound on the entries of matrix H_t. Hence, using this resampling, $H_t(k, j) \leq p$ for all k, j which implies $p(H_t) \leq p$.

[5] To make the data generation possible, for $p \leq 0.3$, we set the parameters of the Dirichlet distribution for the columns of H_1 to $\frac{1000}{r-1}$, otherwise most columns of H_1 are rejected.

Note that p has to be chosen larger than $\frac{1}{r-1}$ since $H(:,j) \in \Delta^r$ for all j, while the columns of H_1 have at least one zero entry.

Finally, the data matrix X is generated by $X = W_t\, H_t$. In the presence of noise, we use additive Gaussian noise based on a given signal-to-noise ratio (SNR). The variance of the i.i.d. random Gaussian noise given the SNR value is given by $\frac{\sum_{i=1}^{m}\sum_{j=1}^{n} X_{i,j}^2}{10^{(SNR/10)} \times m \times n}$.

Parameters for GFPI. The parameters of GFPI are selected according to Table 2. As mentioned before, GFPI is not too sensitive to the parameter η and we use 0.5 in all experiments. For the parameter λ, as it depends on the noise level, it should be decreased as the noise level increases; recall that $\lambda \to +\infty$ in the noiseless case (Theorem 5.5). The parameter γ influences how the data points are associated to a facet: $X(:,j)$ is associated to the facet parametrized by θ when $|X(:,j)^\top\theta - 1| \leq \gamma$. Hence the larger the noise level, the larger γ should be, since the data points are moved further away from the facets.

Table 2

Parameters of GFPI with respect to different values of SNR

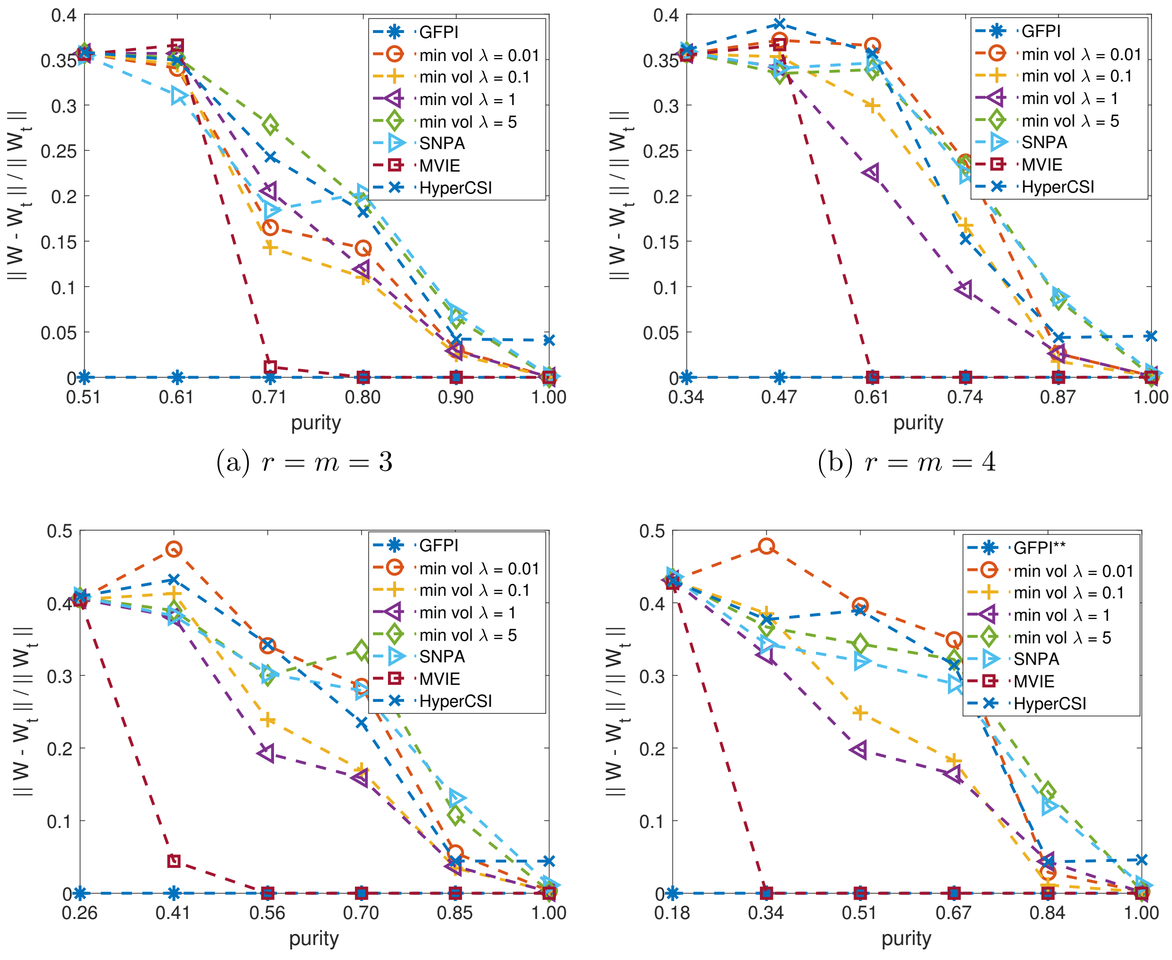
	inf	80	60	50	40	30
λ	1000	100	100	10	10	10
γ	0.001	0.01	0.01	0.05	0.1	0.2

For GFPI, we have set the "timelimit" property of CPLEX to 10 seconds. Whenever the upper bound on CPU time is activated, we specify it with "**" after GFPI in the figures.

6.1.1. Noiseless data sets. In this section, we investigate the effect of the purity on the performance of GFPI compared to the state-of-the-art approaches. To this end, we use the synthetic data with the following parameters: $n_1 = 30$ and $n_2 = 10$. Figure 5 reports the average measure ERR over 10 randomly generated synthetic data sets obtained by the different algorithms for $r = m = \{3, 4, 5, 7\}$ as a function of the purity p. In this experiment, the value of the purity p varies between $\frac{1}{r-1} + 0.01$ (recall, $\frac{1}{r-1}$ is the smallest possible value) to 1 (separability).

GFPI recovers W_t perfectly for all cases, and the performance is not dependent on the purity, as expected since Assumption 3.1 is satisfied, regardless of the purity (Theorem 5.5). On the other hand, the performance of all other approaches gradually decreases as the purity decreases. For SNPA (which is based on the separability assumption), the performance worsens as soon as $p < 1$. For low levels of purity, the SSC is not satisfied, and hence the performances of min vol and MVIE degrade as p decreases. In fact, it is interesting to observe that MVIE performs perfectly for p sufficiently large, when the SSC is satisfies (as guaranteed by the theory), while min vol degrades its performances faster as it relies on local optimization schemes and hence is sensitive to initialization. In fact, initializing min vol with slightly perturbed versions of the groundtruth W leads to rather different solutions with almost perfect recovery. A similar behavior was already observed in [34, Figure 5].

The computational time of the tested algorithms is reported in Table 3. In addition to the running time of GFPI when requiring CPLEX to obtain a global optimality guarantee, Table 3 also reports the time that CPLEX needs to find the optimal solution (before providing the optimality certificate), which we denote GFPI*. We observe that CPLEX finds an optimal



(a) $r = m = 3$ (b) $r = m = 4$

(c) $r = m = 5$ (d) $r = m = 7$

Figure 5. *Average ERR metric for 10 trials depending on the purity for SSMF algorithms in noiseless conditions for different values of r and m.*

solution rather fast, but takes a significant amount of time to provide a certificate of global optimality (this issue is also discussed in Section 5.7). Hence in practice we recommend to use CPLEX with a time limit, as we will do for the numerical experiments on the large-scale hyperspectral images presented in Section 6.2.

Additional numerical experiments regarding the computational time of GFPI can be found in Section SM1.3 of the supplementary material.

6.1.2. Noisy data sets. In this section, we compare the behavior of the different algorithms in the presence of noise. We use two levels of noise (SNR = 60 and 40) and investigate the effect of the purity for $r = m = \{3, 4\}$. Figure 6 reports the ERR metric, similarly as for Figure 5 (average of 10 randomly generated synthetic data sets). As the noise level increases (SNR decreases), the performance of all algorithms decreases steadily. However, in almost all cases, GFPI outperforms all other approaches, especially when the the purity p is low. As for

Table 3

Comparison of the the run times (in seconds) of the tested SSMF algorithms. The experimental setting is the one from Figure 5, with an average over 10 trials. GFPI refers to the time CPLEX needs to find the r optimal solutions (one for each facet), while GFPI refers to the time CPLEX needs to provide a global optimality certificate for these solutions.*

r	purity	GFPI	GFPI*	min vol	SNPA	MVIE	HyperCSI
	0.51	0.64	0.12	0.07	0.008	1.17	0.008
3	0.706	0.64	0.13	0.07	0.008	0.95	0.002
	1	1.02	0.16	0.08	0.008	1.07	0.001
	0.343	2.36	0.41	0.11	0.01	1.77	0.01
4	0.606	4.76	0.53	0.11	0.01	1.87	0.003
	1	6.17	0.39	0.10	0.01	1.58	0.002
	0.26	7.08	0.81	0.14	0.02	5.77	0.009
5	0.556	36.36	0.98	0.13	0.02	7.17	0.004
	1	83.10	0.86	0.11	0.02	5.73	0.003
	0.21	24.73	1.19	0.17	0.04	37.92	0.01
6	0.526	474.92	1.29	0.15	0.04	54.22	0.004
	1	2699.9	1.44	0.14	0.03	40.93	0.003

the noiseless case, MVIE performs the second best. The performance of GFPI in presence of noise and under low purity levels is further illustrated in Section SM1.1.

6.1.3. Rank-deficient SSMF. An advantage of GFPI is that it provably works when W does not have full column rank, and without the separability assumption. Note that

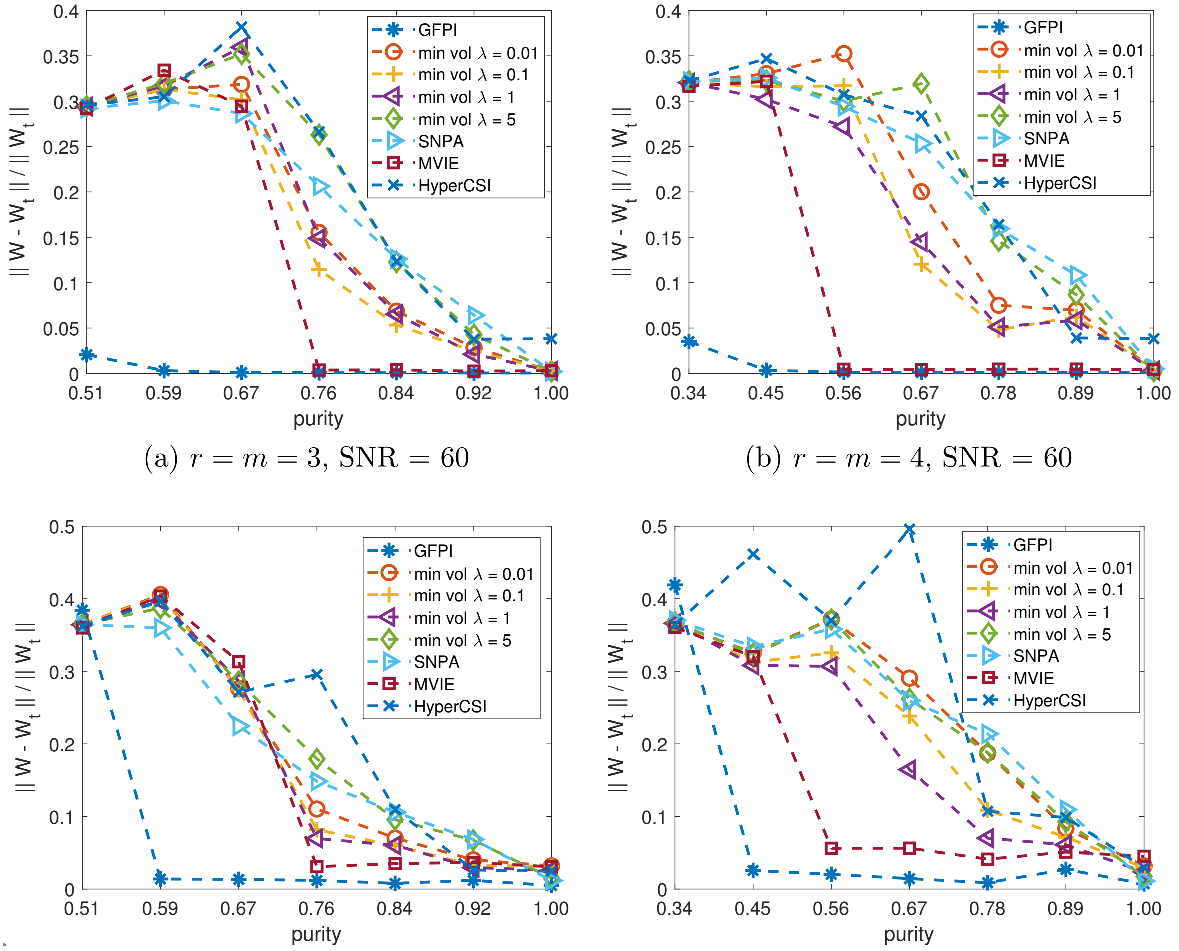
- SNPA works in the rank-deficient case, but requires the separability assumption. Other separable NMF algorithms also work in the rank-deficient case; for example [4, 37, 24] but are computationally much more demanding than SNPA as they rely on solving n linear programs in n variables.
- The min-vol model (6.1) can be used in the rank-deficient case [29]. However, it does not come with identifiability guarantees (this is actually an open problem).

MVIE and HyperCSI are not applicable when $\mathrm{rank}(W) < r$.

In this section, we confirm the ability of GFPI to recover W when it does not have full column rank. To do so, we use the rank-deficient synthetic data from [29]. It generates the matrix $X \in \mathbb{R}^{4\times 200}$ using the rank-deficient matrix

$$W_t = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix},$$

for which $\mathrm{rank}(W_t) = 3 < r = 4$. Each column of $H_t \in \mathbb{R}^{4\times 200}$ is generated using the Dirichlet distribution with parameters equal to 0.1. The columns of H with elements larger than a predefined purity value p are resampled, as before. In this experiment, we consider three values for the purity, namely 0.8, 0.7 and 0.6. We take $X = W_t H_t$ and then corrupt it with i.i.d. Gaussian distribution with zero mean and standard deviation of 0.01. GFPI parameters are $\lambda = 10$, $\eta = 0.5$, $\gamma = 0.05$, and $A = 10$.



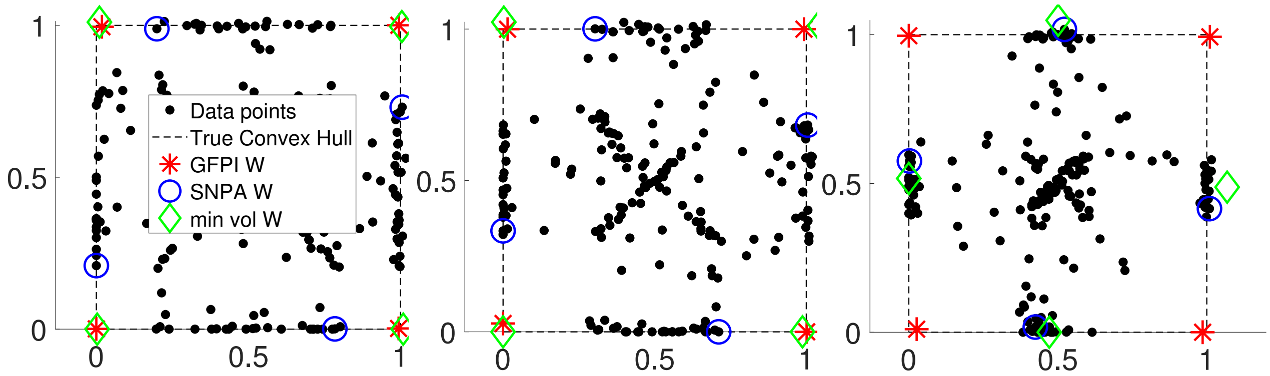
(a) $r = m = 3$, SNR $= 60$ (b) $r = m = 4$, SNR $= 60$

(e) $r = m = 3$, SNR $= 40$ (f) $r = m = 4$, SNR $= 40$

Figure 6. *Average ERR metric for 10 randomly generated data sets depending on purity for the different SSMF algorithms, for different noise levels: SNR of 60 (top) and 40 (bottom), and for $m = r = 3$ (left) and $m = r = 4$ (right).*

Figure 7 shows the result, after projection of the data points in two dimensions. Since the data is not separable, SNPA provides the worst solutions. For $p \in \{0.7, 0.8\}$, min vol performs well, although slightly worse than GFPI; for $p = 0.8$ (resp. 0.7), the ERR of min vol is 0.014 (resp. 0.029) while for GFPI it is 0.010 (resp. 0.018). For $p = 0.6$, min vol fails to extract columns of W_t, as the purity is not large enough. However, it recovers a reasonable solution with smaller volume; this is a similar behavior as in Figure 2.

6.1.4. Performance in the presence of outliers. As mentioned earlier, as far as we know, most SSMF algorithms are very sensitive to outliers (in particular, most separable NMF algorithms, min vol, MVIE and HyperCSI). We generate the clean data by considering $m = r = 3$, $p = 1$ (no resample of the columns of H_t so $p(H_t)$ is close to 1), $n_1 = 30$, $n_2 = 10$ data points (for a total of 100 clean samples), and SNR $= \infty$. We then add outliers whose



(a) purity = 0.8 (b) purity = 0.7 (c) purity = 0.6

Figure 7. *Two dimensional representation of the estimated vertices in rank-deficient cases with different values of purity.*

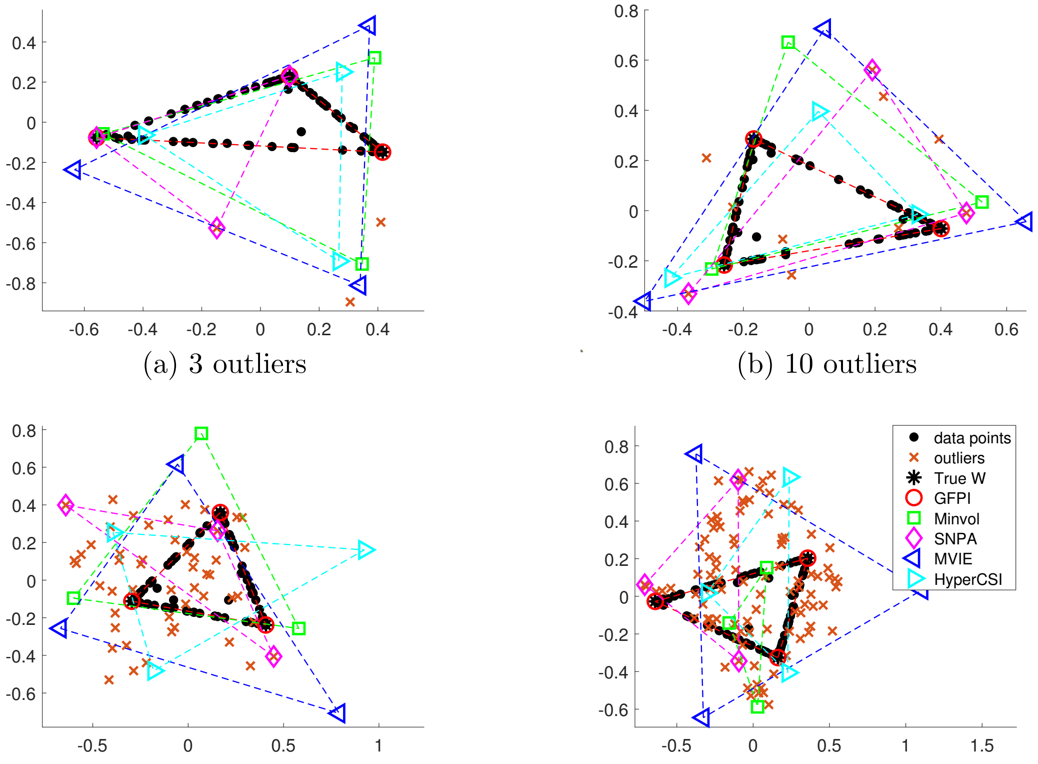
entries are drawn from the uniform distribution in $[0, 1]$. GFPI parameters are $\lambda = 0.01$, $\eta = 0.5$, $\gamma = 0.01$, and $A = 100$. The parameter λ is chosen relatively small allowing δ to take larger values, which is necessary in the presence of outliers. Figure 8 reports the results on four different examples, with 3, 10, 50 and 100 outliers (red crosses). It shows the columns of W and their corresponding convex hulls estimated by the different algorithms. In all cases, GFPI perfectly recovers the true endmembers, while the other algorithms fail. In fact, even few outliers affects their performance whereas GFPI tolerates as many outliers as the number of clean samples. The reason for this robustness to outliers is that outliers are generated randomly, and hence no more than $d - 1$ outliers belong to the same hyperplane (with probability one); in this example, no combination of three outliers belong to the same segment. Of course, adding adversarial outliers on the same hyperplane would lead to different results. However, as long as the number of outliers on the same hyperplane is smaller than the number of points on the facets of $\mathrm{conv}(W)$, GFPI will perform well.

6.2. Hyperspectral images. In this section, we evaluate the performance of GFPI on two widely used hyperspectral images, namely Samson and Jasper Ridge; see [43] and the references therein. These hyperspectral images are relatively large, containing thousands of pixels. Hence we set the *timelimit* of CPLEX for optimizing each facet to 100 seconds. We will provide the MRSA for the extracted factors by the different SSMF algorithms. It is important to note that the ground truth factor W_t is actually unknown, and these estimates come from [43]. Moreover, the reported result for min vol are the best possible performance with highly tuned parameters from [1]. Given W, we solve

$$\min_{H \in \mathbb{R}^{r \times n}} ||X - WH||_F^2 \quad \text{such that} \quad H(:, j) \in \Delta^r \text{ for all } j, \tag{6.2}$$

to estimate the abundance matrix H using the code from [21].

6.2.1. Samson. The Samson data set consists of 95×95 images for 156 spectral bands [43]. Mostly three materials are present in this image: "soil", "water" and "tree", and hence $r = 3$. We run GFPI to extract three endmembers with parameters: $T = d = 3$, $\lambda = 0.1$, $\gamma = 0.3$,



(a) 3 outliers (b) 10 outliers

(c) 50 outliers (d) 100 outliers

Figure 8. *Comparison of SSMF algorithms in the presence of outliers.*

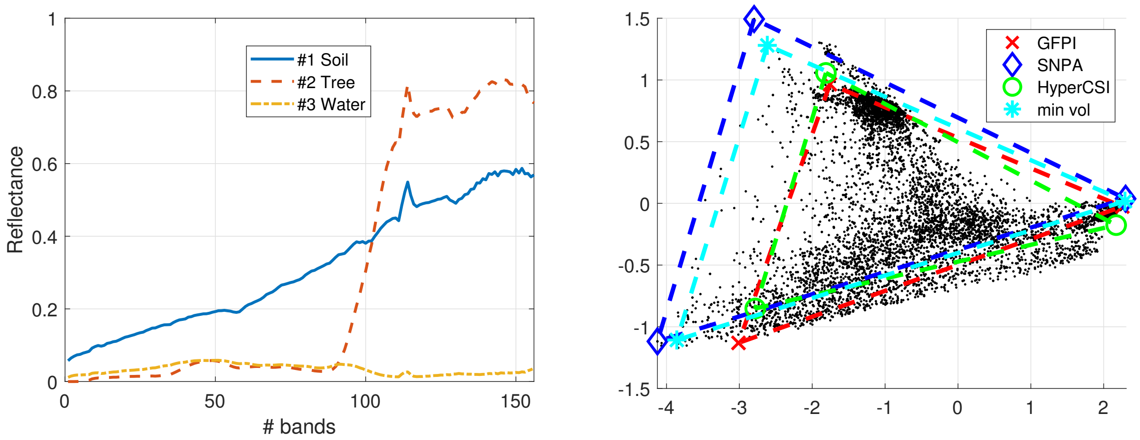
$\eta = 0.7$ and $A = 10$. The extracted spectral signatures are shown in Figure 9 (a). For a qualitative comparison, the corresponding abundance maps are shown in Figure SM5 in the supplementary material. To interpret GFPI geomerically, Figure 9 (b) shows the data points and the polytope computed by GFPI, projected onto a two-dimensional subspace spanned by the first two principal components of the input matrix. Table 4 reports the MRSA and RE for GFPI, SNPA, min vol, and HyperCSI. MVIE is computationally too expensive and is excluded from the comparison. GFPI performs similarly to SNPA and slightly worse than min vol. HyperCSI has the worst performance among the four. This illustrates that CPLEX finds good feasible solutions for the MIP (5.4) fast.

Table 4

Comparing the performances of GFPI with HyperCSI, SNPA and min vol on Samson data set

	SNPA	min vol	HyperCSI	GFPI
MRSA	2.78	2.24	12.91	2.97
$\frac{\|\|X-WH\|\|_F}{\|\|X\|\|_F}$	4.00%	2.64%	5.35%	4.02%

6.2.2. Jasper Ridge. The Jasper Ridge data set consists of 100×100 images for 224 spectral bands [43]. Mostly four materials are present in this image: “road”, “soil”, “water”



(a) Estimated spectral signatures of the three endmembers extracted by GFPI.

(b) Data points (dots), and the estimated endmembers in a 2-D subspace.

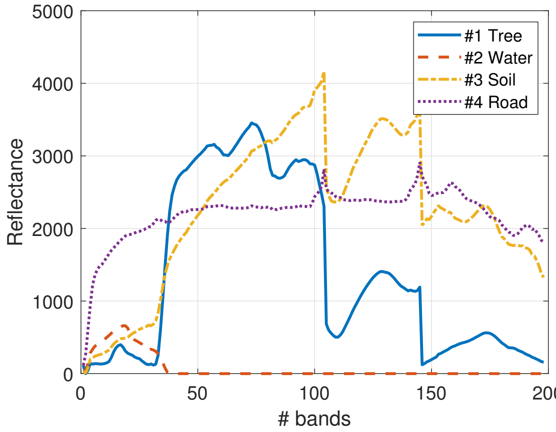
Figure 9. *SSMF algorithms applied on the Samson hyperspectral image.*

and "tree". We run GFPI to extract four endmembers with parameters: $T = d = 4$, $\lambda = 0.0001$, $\gamma = 0.2$, $\eta = 0.5$ and $A = 10$. Note that λ is rather small, much smaller than for Samson ($\lambda = 0.1$). Because such data sets are very noisy and violate the model assumptions, GFPI is more sensitive to its parameters which should be carefully tuned (note that it is also sensitive to the time limit used in CPLEX, and hence to the power of the computer it is run on). However, although GFPI parameters were fine-tuned for these real-world experiments, it provides good solutions for a different values of the parameters. For example, we also obtain good solutions for $\lambda \in [0.01, 0.0001]$. The extracted spectral signatures are shown in Figure 10 (a) and the corresponding abundance maps are reported in Figure SM6. Similar to the Samson data set, the two dimensional representation of the data points and the estimated polytope are shown in Figure 10 (b). Table 5 reports the MRSA and RE. We observe that GFPI has the lowest (best) MRSA value and second best RE among the four algorithms.

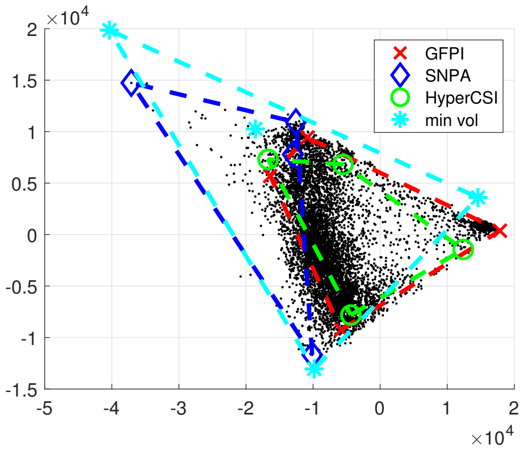
Table 5
Comparing the performances of GFPI with HyperCSI, SNPA and min vol on Jasper database

	SNPA	min vol	HyperCSI	GFPI
MRSA	22.27	6.85	17.04	4.82
$\frac{\|\|X-WH\|\|_F}{\|\|X\|\|_F}$	8.42%	3.90%	11.43%	6.47%

Note that it is natural for min vol to have the lowest RE as it is part of its objective function. Having a low RE for GFPI is a side result of W being well estimated. In particular, GFPI is able to discard outliers (see Section 6.1.4) which may increase the RE significantly because this measure is very sensitive to outliers (least squares). Once W is estimated by GFPI, the RE, or other quality measures, could be used to assess whether GFPI provided a reasonable solution (in fact, GFPI never uses this quantity as a criterion for estimating W). This would be another way to fine tune the parameters of GFPI.



(a) Estimated spectral signatures of the four endmembers extracted by GFPI.



(b) Data points (dots), and the estimated endmembers in a 2-D subspace.

Figure 10. *SSMF algorithms applied on the Jasper ridge hyperspectral image.*

7. Conclusion. In this paper, we have presented a new framework for simplex-structured matrix factorization (SSMF). The high level idea is to identify the facets of the convex hull of the basis matrix W by looking for facets of the convex hull of the data matrix $X = WH$ containing the largest number of points. We first proved that under our facet-based conditions (FBC, see Assumption 3.1), SSMF is identifiable, that is, it has a unique solution W, up to permutation of the columns (Theorem 3.4). Then, we proposed and analyzed brute-force facet-based polytope identification (BFPI) which converts the problem of searching for the facets to the problem of identifying the vertices in the dual space. BFPI recovers the ground truth W under the FBC (Theorem 4.4). We also proposed GFPI (greedy FPI) which sequentially identifies the facets (instead of identifying them all) using MIPs, and comes with identifiabiliy guarantees (Theorem 5.5). In order to handle noise and outliers, we have proposed a very effective MIP to tackle the subproblem for identifying a facet. We have also proposed an effective postprocessing step to improve the recovery of W by reestimating the facets using the data points associated to them. We illustrated the effectiveness of GFPI compared to state-of-the-art SSMF algorithms. GFPI is able to handle highly mixed data points for which the conditions under which the other algorithm work are highly violated (namely, separability and the SSC). It is also able to handle many outliers, and rank-deficient matrices W. We also provided encouraging numerical experiments on real-world hyperspectral images. GFPI is applicable to large data sets because the MIPs do not need to be solved up to global optimality: any solution returned by the solver can be used by GFPI to construct a facet.

Directions of further research include the identifiability of GFPI in presence of noise and outliers, the design of more effective MIP formulations to identify the facets, the improvement of the scalability of GFPI for large-scale data sets (for example by designing dedicated algorithms to solve the MIPs), and the use of GFPI for other applications.

Acknowledgments. We thank the anonymous reviewers for their insightful comments that helped us improve the paper.

REFERENCES

[1] A. M. S. Ang and N. Gillis, *Algorithms and comparisons of nonnegative matrix factorizations with volume regularization for hyperspectral unmixing*, IEEE J. Sel. Top. Appl. Earth. Obs. Remote Sens., 12 (2019), pp. 4843–4853.

[2] M. C. U. Araújo, T. C. B. Saldanha, R. K. H. Galvao, T. Yoneyama, H. C. Chame, and V. Visani, *The successive projections algorithm for variable selection in spectroscopic multicomponent analysis*, Chemometrics and Intelligent Laboratory Systems, 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 International Conference on Machine Learning, 2013, pp. 280–288.

[4] S. Arora, R. Ge, R. Kannan, and A. Moitra, *Computing a nonnegative matrix factorization–provably*, in Proceedings of the forty-fourth annual ACM symposium on Theory of computing, 2012, pp. 145–162.

[5] S. Arora, R. Ge, R. Kannan, and A. Moitra, *Computing a nonnegative matrix factorization—provably*, SIAM Journal on Computing, 45 (2016), pp. 1582–1611.

[6] S. Arora, R. Ge, and A. Moitra, *Learning topic models–going beyond svd*, in IEEE 53rd Annual Symposium on Foundations of Computer Science, IEEE, 2012.

[7] P. Belotti, P. Bonami, M. Fischetti, A. Lodi, M. Monaci, A. Nogales-Gómez, and D. Salvagnin, *On handling indicator constraints in mixed integer programming*, Computational Optimization and Applications, 65 (2016), pp. 545–566.

[8] J. M. Bioucas-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. Top. Appl. Earth. Obs. Remote Sens., 5 (2012), pp. 354–379.

[9] D. Bremner, K. Fukuda, and A. Marzetta, *Primal-dual methods for vertex and facet enumeration*, Discrete & Computational Geometry, 20 (1998), pp. 333–357.

[10] A. Cichocki, R. Zdunek, A. H. Phan, and S.-i. Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*, John Wiley & Sons, 2009.

[11] J. E. Cohen and N. Gillis, *Identifiability of complete dictionary learning*, SIAM Journal on Mathematics of Data Science, 1 (2019), pp. 518–536.

[12] CPLEX IBM ILOG, *V12.1: Users manual for CPLEX*, International Business Machines Corporation, 46 (2009), p. 157.

[13] X. Fu, K. Huang, and N. D. Sidiropoulos, *On identifiability of nonnegative matrix factorization*, IEEE Signal Processing Letters, 25 (2018), pp. 328–332.

[14] X. Fu, K. Huang, N. D. Sidiropoulos, and W.-K. Ma, *Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications.*, IEEE Signal Processing Magazine, 36 (2019), pp. 59–80.

[15] X. Fu, K. Huang, N. D. Sidiropoulos, Q. Shi, and M. Hong, *Anchor-free correlated topic modeling*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 41 (2019), pp. 1056–1071.

[16] X. Fu, K. Huang, B. Yang, W.-K. Ma, and N. D. Sidiropoulos, *Robust volume minimization-based matrix factorization for remote sensing and document clustering*, IEEE Transactions on Signal Processing, 64 (2016), pp. 6254–6268.

[17] X. Fu, W.-K. Ma, T.-H. Chan, and J. M. Bioucas-Dias, *Self-dictionary sparse regression for hyperspectral unmixing: Greedy pursuit and pure pixel search are related*, IEEE Journal of Selected Topics in Signal Processing, 9 (2015), pp. 1128–1141.

[18] X. Fu, W.-K. Ma, K. Huang, and N. D. Sidiropoulos, *Blind separation of quasi-stationary sources: Exploiting convex geometry in covariance domain*, IEEE Transactions on Signal Processing, 63 (2015), pp. 2306–2320.

[19] X. Fu, N. Vervliet, L. De Lathauwer, K. Huang, and N. Gillis, *Computing large-scale matrix and tensor decomposition with structured factors: A unified nonconvex optimization perspective*, IEEE Signal Processing Magazine, 37 (2020), pp. 78–94.

[20] R. Ge and J. Zou, *Intersecting faces: Non-negative matrix factorization with new guarantees*, in Proceedings of the 32nd International Conference on Machine Learning, 2015, pp. 2295–2303.

[21] N. Gillis, *Successive nonnegative projection algorithm for robust nonnegative blind source separation*, SIAM Journal on Imaging Sciences, 7 (2014), pp. 1420–1450.

[22] N. Gillis, *The why and how of nonnegative matrix factorization*, Regularization, Optimization, Kernels, and Support Vector Machines, 12 (2014).

[23] N. Gillis and A. Kumar, *Exact and heuristic algorithms for semi-nonnegative matrix factorization*, SIAM Journal on Matrix Analysis and Applications, 36 (2015), pp. 1404–1424.

[24] N. Gillis and R. Luce, *Robust near-separable nonnegative matrix factorization using linear optimization*, Journal of Machine Learning Research, 15 (2014), pp. 1249–1280.

[25] N. Gillis and S. A. Vavasis, *Fast and robust recursive algorithms for separable nonnegative matrix factorization*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 36 (2014), pp. 698–714.

[26] K. Huang, X. Fu, and N. D. Sidiropoulos, *Learning hidden markov models from pairwise co-occurrences with applications to topic modeling*, arXiv preprint arXiv:1802.06894, (2018).

[27] K. Huang, N. D. Sidiropoulos, and A. Swami, *Non-negative matrix factorization revisited: Uniqueness and algorithm for symmetric decomposition*, IEEE Transactions on Signal Processing, 62 (2014), pp. 211–224.

[28] M. Kleder, *Con2vert-constraints to vertices*, MathWroks File Exchange. Available at https://au. mathworks. com/matlabcentral/fileexchange, (2005).

[29] V. Leplat, A. M. Ang, and N. Gillis, *Minimum-volume rank-deficient nonnegative matrix factorizations*, in IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 3402–3406.

[30] V. Leplat, N. Gillis, and A. M. S. Ang, *Blind audio source separation with minimum-volume beta-divergence NMF*, IEEE Transactions on Signal Processing, 68 (2020), pp. 3400–3410.

[31] C.-H. Lin and J. M. Bioucas-Dias, *Nonnegative blind source separation for ill-conditioned mixtures via John ellipsoid*, IEEE Transactions on Neural Networks and Learning Systems, (2020).

[32] C.-H. Lin, C.-Y. Chi, Y.-H. Wang, and T.-H. Chan, *A fast hyperplane-based minimum-volume enclosing simplex algorithm for blind hyperspectral unmixing*, IEEE Transactions on Signal Processing, 64 (2015), pp. 1946–1961.

[33] C.-H. Lin, W.-K. Ma, W.-C. Li, C.-Y. Chi, and A. Ambikapathi, *Identifiability of the simplex volume minimization criterion for blind hyperspectral unmixing: The no-pure-pixel case*, IEEE Trans. Geosci. Remote Sens., 53 (2015), pp. 5530–5546.

[34] C.-H. Lin, R. Wu, W.-K. Ma, C.-Y. Chi, and Y. Wang, *Maximum volume inscribed ellipsoid: A new simplex-structured matrix factorization framework via facet enumeration and convex optimization*, SIAM Journal on Imaging Sciences, 11 (2018), pp. 1651–1679.

[35] W.-K. Ma, J. M. Bioucas-Dias, T.-H. Chan, N. Gillis, P. Gader, A. J. Plaza, A. Ambikapathi, and C.-Y. Chi, *A signal processing perspective on hyperspectral unmixing: Insights from remote sensing*, IEEE Signal Processing Magazine, 31 (2013), pp. 67–81.

[36] L. Miao and H. Qi, *Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization*, IEEE Trans. Geosci. Remote Sens., 45 (2007), pp. 765–777.

[37] B. Recht, C. Re, J. Tropp, and V. Bittorf, *Factoring nonnegative matrices with linear programs*, in Advances in Neural Information Processing Systems, 2012, pp. 1214–1222.

[38] G. Salmani Jajaei, *Rotating Supporting Hyperplanes and Snug Circumscribing Simplexes*, PhD thesis, Virginia Commonwealth University, 2018.

[39] Y. Sun and J. Xin, *Underdetermined sparse blind source separation of nonnegative and partially overlapped data*, SIAM Journal on Scientific Computing, 33 (2011), pp. 2063–2094.

[40] M. Udell, C. Horn, R. Zadeh, and S. Boyd, *Generalized low rank models*, Foundations and Trends in Machine Learning, 9 (2016), pp. 1–118.

[41] L. A. Wolsey, *Mixed integer programming*, Wiley Encyclopedia of Computer Science and Engineering, (2007), pp. 1–10.

[42] R. Wu, W.-K. Ma, and X. Fu, *A stochastic maximum-likelihood framework for simplex structured matrix factorization*, IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), (2017), pp. 2557–2561.

[43] F. Zhu, *Hyperspectral unmixing: ground truth labeling, datasets, benchmark performances and survey*, arXiv preprint arXiv:1708.05125, (2017).

[44] G. Ziegler, *Lectures on Polytopes*, Springer-Verlag, 1995.