

---

# Intersecting Faces: Non-negative Matrix Factorization With New Guarantees

---

**Rong Ge**

Microsoft Research New England

RONGGE@MICROSOFT.COM

**James Zou**

Microsoft Research New England

JAZO@MICROSOFT.COM

## Abstract

Non-negative matrix factorization (NMF) is a natural model of admixture and is widely used in science and engineering. A plethora of algorithms have been developed to tackle NMF, but due to the non-convex nature of the problem, there is little guarantee on how well these methods work. Recently a surge of research have focused on a very restricted class of NMFs, called separable NMF, where provably correct algorithms have been developed. In this paper, we propose the notion of subset-separable NMF, which substantially generalizes the property of separability. We show that subset-separability is a natural necessary condition for the factorization to be unique or to have minimum volume. We developed the Face-Intersect algorithm which provably and efficiently solves subset-separable NMF under natural conditions, and we prove that our algorithm is robust to small noise. We explored the performance of Face-Intersect on simulations and discuss settings where it empirically outperformed the state-of-art methods. Our work is a step towards finding provably correct algorithms that solve large classes of NMF problems.

## 1. Introduction

In many settings in science and engineering the observed data are admixtures of multiple latent sources. We would typically want to infer the latent sources as well as the admixture distribution given the observations. Non-negative matrix factorization (NMF) is a natural mathematical framework to model many admixture problems.

In NMF we are given an observation matrix  $M \in \mathbb{R}^{n \times m}$ , where each row of  $M$  corresponds to a data-point in  $\mathbb{R}^m$ .

---

*Proceedings of the 32<sup>nd</sup> International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37. Copyright 2015 by the author(s).*

We assume that there are  $r$  latent sources, modeled by the unobserved matrix  $W \in \mathbb{R}^{r \times m}$ , where each row of  $W$  characterizes one source. Each observed data-point is a linear combination of the  $r$  sources and the combination weights are encoded in a matrix  $A \in \mathbb{R}^{n \times r}$ . Moreover, in many natural settings, the sources are non-negative and the combinations are additive. The computational problem is then to factor a given matrix  $M$  as  $M = AW$ , where all the entries of  $M$ ,  $A$  and  $W$  are non-negative. We call  $r$  the inner-dimension of the factorization, and the smallest possible  $r$  is usually called the nonnegative rank of  $M$ . NMF was first purposed by (Lee & Seung, 1999), and has been widely applied in computer vision (Lee & Seung, 2000), document clustering (Xu et al., 2003), hyperspectral unmixing (Nascimento & Dias, 2004; Gomez et al., 2007), computational biology (Devarajan, 2009), etc. We give two concrete examples

**Example 1.** In topic modeling,  $M$  is the  $n$ -by- $m$  word-by-document matrix, where  $n$  is the vocabulary size and  $m$  is the number of documents. Each column of  $M$  corresponds to one document and the entry  $M(i, j)$  is the frequency with which word  $i$  appears in document  $j$ . The topics are the columns of  $A$ , and  $A(i, k)$  is the probability that topic  $k$  uses word  $i$ .  $W$  is the topic-by-document matrix and captures how much each topic contributes to each document. Since all the entries of  $M$ ,  $A$  and  $W$  are frequencies, they are all non-negative. Given  $M$  from a corpus of documents, we would like to factor  $M = AW$  and recover the relevant topics in these documents. (Note that in this example  $A$  is the matrix of “sources” and  $W$  is the matrix of mixing weights, so it is the transpose of what we just introduced. We use this notation to be consistent with previous works (Arora et al., 2012).)

**Example 2.** In many bio-medical applications, we collect samples and for each sample perform multiple measurements (e.g. expression of  $10^4$  genes or DNA methylation at  $10^6$  positions in the genome; all the values are non-negative).  $M$  is the sample-by-measurement matrix, where  $M(i, j)$  is the value of the  $j$ th measurement in sam-

ple  $i$ . Each sample, whether taken from humans or animals, is typically a composition of several cell-types that we do not directly observe. Each row of  $W$  corresponds to one cell-type, and  $W(k, j)$  is the value of cell-type  $k$  in measurement  $j$ . The entry  $A(i, k)$  is the fraction of sample  $i$  that consists of cell-type  $k$ . Experiments give us the matrix  $M$ , and we would like to factor  $M = AW$  to identify the relevant cell-types and their compositions in our samples.

Despite the simplicity of its formulation, NMF is a challenging problem. First, the NMF problem may not be identifiable, and hence we can not hope to recover the *true*  $A$  and  $W$ . Moreover, even ignoring the identifiability Vavasis (2009) showed that finding any factorization  $M = AW$  with inner-dimension  $r$  is an *NP*-hard problem. Arora et al. (2012) showed under reasonable assumptions we cannot hope to find a factorization in time  $(mn)^{o(r)}$ , and the best algorithm known is Moitra (2013) that runs in time  $O(2^r mn)^{O(r^2)}$ .

Many heuristic algorithms have been developed for NMF but they do not have guarantees for when they would actually converge to the true factorization (Lee & Seung, 2000; Lin, 2007). More recently, there has been a surge of interest in constructing practical NMF algorithms with strong theoretical guarantees. Most of this activity (e.g. Arora et al. (2012); Bittorf et al. (2012); Kumar et al. (2012); Gillis (2012); Gillis & Vavasis (2014), see more in Gillis (2014)) are based on the notion of *separability* (Donoho & Stodden, 2003) which is a very strict condition that requires that all the rows of  $W$  appear as rows in  $M$ . While this might hold in some document corpus, it is unlikely to be true in other engineering and bio-medical applications.

**Our Results** In this paper, we develop the notion of *subset separability*, which is a significantly weaker and more general condition than separability. In topic models, for example, separability states that there is a word that is unique to each topic. Subset separability means that there is a combination of words that is unique to each topic. We show that subset separability arise naturally as a necessary condition when the NMF is identifiable or when we are seeking the minimal volume factorization. We characterize settings when subset-separable NMF can be solved in polynomial-time, and this include the separable setting as a special case. We construct the Face-Intersect algorithm which provably and robustly solves the NMF even in the presence of adversarial noise. We use simulations to explore conditions where our algorithm achieves more accurate inference than current state-of-art algorithms.

**Organization** We first describe the geometric interpretation of NMF (Sec. 2), which leads us to the notion of subset-separable NMF (Sec. 3). We then develop our Face-Intersect algorithm and analyze its robustness (Sec. 4). Our main result, Theorem 4.2, states that for subset-separable NMF, if the facets are *properly filled* in a way that de-

pends on the magnitude of the adversarial noise, then Face-Intersect is guaranteed to find a factorization that is close to the true factorization in polynomial time. We discuss the algorithm in more detail in Sections 5 and 6, and analyze a generative model that give rise to properly filled facets in Section 7. Finally we present experiments to explore settings where Face-Intersect outperforms state-of-art NMF algorithms (Sec. 8). Due to space constraints, all the proofs are presented in the appendix. Throughout the paper, we give intuitions behind proofs of the main results.

## 2. Geometric intuition

For a matrix  $M \in \mathbb{R}^{n \times m}$ , we use  $M^i \in \mathbb{R}^m$  to denote the  $i$ -th row of  $M$ , but it is viewed as a column vector. Given a factorization  $M = AW$ , without loss of generality we can assume the rows of  $M, A, W$  all sum up to 1 (this can always be done, see (Arora et al., 2012)). In this way we can view the rows of  $W$  as vertices of an unknown simplex, and the rows of  $M$  are all in the convex hull of these vertices. The NMF is then equivalent to the following geometric problem:

**NMF, Geometric Interpretation** There is an unknown  $W$ -simplex whose vertices are the rows of  $W \in \mathbb{R}^m$ ,  $W^1, \dots, W^r$ . We observe  $n$  points  $M^1, M^2, \dots, M^n \in \mathbb{R}^m$  (corresponding to rows of  $M$ ) that lie in the  $W$ -simplex. The goal is to identify the vertices of the  $W$ -simplex.

When clear from context, we also call the  $W$  matrix as the simplex, and the goal is to find the vertices of this simplex. There is one setting where it is easy to identify all the vertices.

**Definition 2.1** (separability). *A NMF is separable if all the vertices  $W^j$ 's appear in the points  $M^i$ 's that we observe.*

Separability was introduced in Donoho & Stodden (2003). When the NMF is separable, the problem simplifies as we only need to identify which of the points  $M^j$ 's are vertices of the simplex. This can be done in time polynomial in  $n, m$  and  $r$  (Arora et al., 2012). Separability is a highly restrictive condition and it takes advantage of only the 0-dimensional structure (vertices) of the simplex. In this work, we use higher dimensional structures of the simplex to solve the NMF. We use the following standard definition of facets:

**Definition 2.2** (facet). *A facet  $S \subset [r]$  of the  $W$ -simplex is the convex hull of vertices  $\{W^j : j \in S\}$ . We call  $S$  a filled facet if there is at least one point  $M^i$  in the interior of  $S$  (or if  $|S| = 1$  and there is one point  $M^i$  that is equal to that vertex; such  $M^i$  is called an anchor).*

**Conventions** When it's clear from context, we interchangeably represent a facet  $S$  both by the indices of its

vertices and by the convex hull of these vertices. A facet also corresponds to a unique linear subspace  $Q_S$  with dimension  $|S|$  that is the span of  $\{W^j : j \in S\}$ . In the rest of the paper, it's convenient to use linear algebra to quantify various geometric ideas. We will represent a  $d$ -dimensional subspace of  $\mathbb{R}^m$  using a matrix  $U \in \mathbb{R}^{m \times d}$ , the columns of matrix  $U$  is an arbitrary orthonormal basis for the subspace (hence the representation is not unique). We use  $P_U = UU^T$  to denote the projection matrix to subspace  $U$ , and  $U^\perp \in \mathbb{R}^{m \times (m-d)}$  to denote an arbitrary representation of the orthogonal subspace. For two subspaces  $U$  and  $V$  of the same dimension, we define their distance to the the sin of the principle angle between the two subspaces (this is the largest angle between vectors  $u, v$  for  $u \in U$  and  $v \in V$ ). This distance can be computed as the spectral norm  $\|P_{U^\perp} V\|$  (and has many equivalent formulations).

### 3. Subset Separability

NMF is not identifiable up to scalings and permutations of the rows of  $W$ . Ignoring such transformations, there can still be multiple non-negative factorizations of the same matrix  $M$ . This arise when there are different sets of  $r$  vertices in the non-negative orthant that contain all the points  $M^i$  in its convex hull. For example, suppose  $M = AW$  and the  $A$  matrix has all positive entries. All the points  $M^i$  are in the interior of the  $W$ -simplex. Then it is possible to perturb the vertices of  $W$  while still maintaining all of the  $M^i$ 's in its convex hull. This give rise to a different factorization  $M = \hat{A}\hat{W}$ . When the factorization is not unique, we may want find a solution where the  $W$ -simplex has minimal volume, in the sense that it is impossible to move a single vertex and shrink the volume while maintaining the validity of the solution.

It's clear that in order for  $W$  to be the minimal volume solution to the NMF, there must be some points  $M^i$  that lie on the boundary of the  $W$ -simplex. We show that a necessary condition for  $W$  to be volume minimizing is for the filled facets (facets of  $W$  with points in its interior) to be subset-separable. Intuitively, this means that each vertex of  $W$  is the unique intersection point of a subset of filled facets.

**Definition 3.1** (subset-separable). *A NMF  $M = AW$  is subset-separable if there is a set of filled facets  $S_1, \dots, S_k \subset [r]$  such that  $\forall j \in [r]$ , there is a subset of  $S_{j_1}, S_{j_2}, \dots, S_{j_{k_j}}$  whose intersection is exactly  $j$ .*

**Proposition 3.1.** *Suppose  $W$  is a minimal volume rank  $r$  solution of the NMF  $M = AW$ . Then  $W$  is subset-separable.*

It is easy to see that the factorization  $M = AW$  is subset-separable is equivalent to the property that for every  $j_1 \neq j_2 \in [r]$ , there is a row  $i$  of  $A$  such that

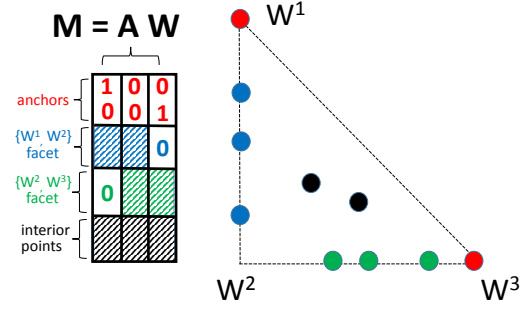


Figure 1. Illustration of the NMF geometry.

$A_{i,j_1} = 0$  and  $A_{i,j_2} \neq 0$ . The previously proposed separability condition corresponds to the special case where the filled facets  $S_1, \dots, S_k$  correspond to the singleton sets  $\{W^1\}, \dots, \{W^r\}$ .

**Example.** We illustrate the subset-separable condition in Figure 1. In this figure, the circles correspond to data points  $M^i$ 's and they are colored according to the facet that they belong. The filled facets are  $S_1 = \{1\}$ ,  $S_2 = \{3\}$ ,  $S_3 = \{1, 2\}$  and  $S_4 = \{2, 3\}$ . The facet  $\{W^1, W^3\}$  is not filled since there are no points in its interior. The singleton facets  $S_1$  and  $S_2$  are also called anchors. This NMF is subset-separable since  $W^2$  is the unique intersection of  $S_3$  and  $S_4$ , but it is not separable. The figure also illustrates the corresponding  $A$  matrix, where the rows are grouped by facets and the shaded entries denote the support of each row.

The geometry of the simplex suggests an intuitive meta-algorithm for solving subset-separable NMFs, which is the basis of our Face-Intersect algorithm.

1. Identify the filled facets,  $S_1, \dots, S_k$ ,  $r \leq k \leq n$ .
2. Take intersections of the facets to recover all the rows of  $W$  (vertices of the simplex).
3. Use  $M$  and  $W$  to solve for  $A$ .

### 4. Robust algorithm for subset-separable NMF

In order to carry out the meta-algorithm, the key computational challenge is to efficiently and correctly identify the filled facets of the  $W$  simplex. Finding filled facets is related to well-studied problems in subspace clustering (Vidal, 2010) and subspace recovery (Hardt & Moitra, 2013). In subspace clustering we are given points in  $k$  different subspaces and the goal is to cluster the points according to which subspace it belong to. This problem is in general NP-hard (Elhamifar & Vidal, 2009) and can only be solved under strong assumptions. Subspace recovery tries to find a unique subspace a fraction  $p$  of the points. Hardt

& Moitra (2013) showed this problem is hard unless  $p$  is large compared to the ratio of the dimensions. Techniques and algorithms from subspace clustering and recovery typically make strong assumptions about the independence of subspaces or the generative model of the points, and cannot be directly applied to our problem. Moreover, our filled facets have the useful property that they are on the boundary of the convex hull of the data points, which is not considered in general subspace clustering/discovery methods. We identified a general class of filled facets, called *properly filled facets* that are computationally efficient to find.

**Definition 4.1** (properly filled facets). *Given a NMF  $M = AW$ , a set of facets  $S_1, \dots, S_k \in [r]$  of  $W$  is properly filled if it satisfies the following properties:*

1. *For any facet  $|S_i| > 1$ , the rows of  $A$  with support equal to  $S_i$  (i.e. points that lie on this facet) has a  $|S_i| - 1$ -dimensional convex hull. Moreover, there is at least one row of  $A$  that is in the interior of the convex hull.*
2. *(General positions property.) For any subspace of dimension  $1 < t < r$ , if it contains more than  $t$  rows in  $M$ , then the subspace contains at least one  $S_i$  which is not a singleton facet.*

Condition 1 ensures that each  $S_i$  has sufficiently many points to be non-degenerate. Condition 2 says that points that are not in the lower dimensional facets  $S_1, \dots, S_k$  are in general positions, so that no random subspace look like a properly filled facet. A set of properly filled facets  $S_1, \dots, S_k$  may contain singleton sets corresponding some of the rows  $W^j$  if these rows also appear as rows in  $M$ . We first state the main results and then state the Face-Intersect algorithm.

**Theorem 4.1.** *Suppose  $M = AW$  is subset separable by  $S_1, \dots, S_k$  and these facets are properly filled, then given  $M$  the Face-Intersect algorithm computes  $A$  and  $W$  in time polynomial in  $n, m$  and  $r$  (and in particular the factorization is unique).*

In many applications, we have to deal with noisy NMF  $\hat{M} = AW + \text{noise}$  where (potentially correlated) noise is added to rows of the data matrix  $M$ . Suppose every row is perturbed by a small noise  $\epsilon$  (in  $\ell_2$  norm), we would like the algorithm to be robust to such additive noise. We need a generalization of properly filled facets.

**Definition 4.2** ( $(N, H, \gamma)$  properly filled facets). *Given a NMF  $M = AW$ , a set of facets  $S_1, \dots, S_k \in [r]$  of  $W$  is  $(N, H, \gamma)$  properly filled if it satisfies the following properties:*

1. *In any set  $|S_i| > 1$ , there is a row  $i^*$  in  $A$  whose support is equal to  $S_i$ , and is in the convex hull*

*of other rows of  $A$ . There exists a convex combination  $M^{i^*} = \sum_{i \in [n] \setminus i^*} w_i M^i$ , such that the matrix  $\sum_{i \in [n] \setminus i^*} w_i (M^i)(M^i)^T$  has rank  $|S_i|$ , and the smallest nonzero singular value is at least  $\gamma$ . We call this special point  $M^{i^*}$  the center for this facet.*

2. *For any set  $|S_i| > 1$ , there are at least  $N$  rows in  $A$  whose support is exactly equal to  $S_i$ .*
3. *For any subspace  $Q$  of dimension  $1 < t < r$ , if there are at least  $N$  rows of  $M$  in an  $\epsilon$ -neighborhood of  $Q$ , then there exists a non-singleton set  $S_i$  with corresponding subspace  $Q_i$  such that  $\|P_{Q^\perp} Q_i\| \leq H\epsilon$ .*

Intuitively, if we represent the center point as a convex combination of other points, the only points that have a nonzero contribution must be on the same facet as the center. Condition 1 then ensures there is a “nice” convex combination that allows us to robustly recover the subspace corresponding to the facet even in presence of noise. Condition 2 shows every properly filled facets contain many points, which is why they are different from other subspaces and are the facets of the true solution. Condition 3 is a generalization of the general position property, which essentially says “every subspace that contains many points must be close to a properly filled facet”. In Section 7 we show that under a natural generative model, the NMF has  $(N, H, \gamma)$ -properly filled facets with high probability.

Properly filled facets is a property of how the points  $M^i$  are distributed on the facets of  $W$ . The geometry of the  $W$ -simplex itself also affects the accuracy of our Face-Intersect algorithm.

**Definition 4.3.** *A matrix  $W \in \mathbb{R}^{r \times m}$  ( $r \leq m$ ) is  $\alpha$ -robust if its rows have norm bounded by 1, and its  $r$ -th singular value is at least  $\alpha$ .*

Under these assumptions we prove that Face-Intersect robustly learns the unknown simplex  $W$ .

**Theorem 4.2.** *Suppose  $M = AW$  is subset separable by  $S_1, \dots, S_k$  and these facets are  $(N, H, \gamma)$  properly filled, and the matrix  $W$  is  $\alpha$ -robust. Then given  $\hat{M}$  whose rows are within  $\ell_2$  distance  $\epsilon$  to  $M$ , with  $\epsilon < o(\alpha^4 \gamma / Hr^3)$ , Algorithm Face-Intersect finds  $\hat{W}$  such that there exists a permutation  $\pi$  and for all  $i$   $\|\hat{W}_i - W_{\pi(i)}\| \leq O(Hr^2 \epsilon / \alpha^2 \gamma)$ . The running time is polynomial in  $n, m$  and  $r$ .*

A vertex  $j \in [r]$  is an intersection vertex if there exists a subset of properly filled facets  $\{S_{j_k} : |S_{j_k}| \geq 2\}$  such that  $j = \cap_k S_{j_k}$ . Since the first module of Face-Intersect, Algorithm 3, only finds non-singleton facets, the intersection vertices are all the vertices that we could find using these facets. The last module of Face-Intersect finds all the remaining vertices of the simplex.

---

**Algorithm 1** Face-Intersect

Run Algorithm 3 to find subspaces that correspond to properly filled facets  $S_1, S_2, \dots, S_k$  where  $|S_i| \geq 2$ .  
 Run Algorithm 5 to find the *intersection vertices*  $P$ .  
 Run Algorithm 5 (similar to Algorithm 4 in (Arora et al., 2013)) to find the singleton points (anchors).  
 Given  $\hat{M}, \hat{W}$ , compute  $\hat{A}$ .

---

**Our approach** The main idea of our algorithm is to first find the subspaces corresponding properly filled facets, then take the intersections of these facets to find the intersection vertices. Finally we adapt the algorithm from (Arora et al., 2013) to find the remaining vertices that correspond to singleton sets.

- **Finding facets** For each row of  $M$ , we try to represent it as the convex combination of other rows of  $M$ . We use an iterative algorithm to make sure the span of points used in this convex combination is exactly the subspace corresponding to the facet.
- **Removing false positives** The previous step will generate subspaces that correspond to properly filled facets, but it might also generate false positives (subspaces that do not correspond to any properly filled facets). Condition 3 in Definition 4.2 allows us to filter out these false positives as these subspaces will not contain enough nearby points.
- **Finding intersection vertices** We design an algorithm that systematically tries to take the intersections of subspaces in order to find the intersection vertices. This relies on the subset-separable property and robustness properties of the simplex. This step computes at most  $O(nr)$  subspace intersection operations.
- **Finding remaining vertices** The remaining vertices correspond to the singleton sets. This is similar to the separable case and we use an algorithm from Arora et al. (2013).

## 5. Finding properly filled facets

In this section we show how to find properly filled facets  $S_i$  with  $|S_i| \geq 2$ . The singleton facets (anchors) are not considered in this section, since they will be found through a separate algorithm. We first show how to find a properly filled facet if we know its center (Condition 1 in Definition 4.2). Then to find all the properly filled facets we enumerate points to be the center and remove false positives.

**Finding one properly filled facet** Given the center point, if there is no noise then when we represent this point as convex combinations of other points, all the points with positive weight will be on the same facet. Intuitively the span of these points should be equal to the subspace cor-

responding to the facet. However there are two key challenges here: first we need to show that when there is noise, points with large weights in the convex combination are close to the true facet; second, it is possible that points with large weights only span a lower dimensional subspace of the facet. Condition 1 in Definition 4.2 guarantees that there exists a *nice* convex combination that spans the entire subspace (and robustly so because the smallest singular value is large compared to noise). In Algorithm 2, we iteratively improve our convex combination and eventually converge to this nice combination.

---

**Algorithm 2** Finding a properly filled facet

**input** points  $\hat{v}^1, \hat{v}^2, \dots, \hat{v}^n$ , and center point  $\hat{v}^0$  (Condition 1 in Definition 4.2).

**output** the proper facet containing  $\hat{v}^0$ .

- 1: Maintain a subspace  $\hat{Q}$  (initially empty)
- 2: Iteratively solve the following optimization program:

$$\begin{aligned} \max \quad & \text{tr}(P_{\hat{Q}^\perp} \sum_{i=1}^n w_i \hat{v}^i (\hat{v}^i)^T P_{\hat{Q}^\perp}) \\ & \forall i \in [n] \quad w_i \geq 0 \\ & \sum_{i=1}^n w_i = 1 \\ & \|\hat{v}^0 - \sum_{i=1}^n w_i \hat{v}^i\| \leq 2\epsilon \\ \text{diag}(\hat{Q}^T \left( \sum_{i=1}^n w_i \hat{v}^i (\hat{v}^i)^T \right) \hat{Q}) & \geq \gamma/2. \end{aligned}$$

- 3: Let  $\hat{Q}$  be the top singular space of  $(\sum_{i=1}^n w_i \hat{v}^i (\hat{v}^i)^T)$  for singular values larger than  $\gamma/2d$ .
  - 4: Repeat until the dimension of  $\hat{Q}$  does not increase.
- 

**Theorem 5.1.** Suppose  $\|\hat{v}^i - v^i\| \leq \epsilon$ ,  $v^0$  is the center point of a properly filled facet  $S \subset [r]$  with  $|S| = d$ , and the unknown simplex  $W$  is  $\alpha$ -robust, when  $d\sqrt{r}\epsilon/\alpha\gamma \ll 1$  Algorithm 2 stops within  $d$  iterations, and the subspace  $\hat{Q}$  is within distance  $O(\sqrt{r}\epsilon/\alpha\gamma)$  to the true subspace  $Q_S$ .

The intuition of Algorithm 2 is to maintain a convex combination for the center point. We show for any convex combination, the top singular space associated with the combination,  $\hat{Q}$ , is always close to a subspace of the true space  $Q_S$ . The algorithm then tries to explore other directions by maximizing the projection that is outside the current subspace  $\hat{Q}$  (the objective function of the convex optimization), while maintaining that the current subspace have large singular values (the last constraint). In the proof we show since there is a nice solution, the algorithm will always be able to make progress until the final solution is a *nice* convex combination.

**Finding all subsets** Algorithm 2 can find one properly filled facet, if we have its center point (Condition 1 in Definition 4.2). In order to find all the properly filled facets, we enumerate through rows of  $M$  and prune false positives using Condition 3 in Definition 4.2

---

**Algorithm 3** Finding all proper facets
 

---

**input**  $\hat{M}$  whose factorization is subset-separable with  $(N, H, \gamma)$ -properly filled facets.

- 1: **for**  $i = 1$  to  $n$  **do**
- 2:   Let  $\hat{v}^0 = \hat{M}^i$  and  $\hat{v}^1, \dots, \hat{v}^{n-1}$  be the rest of vertices.
- 3:   Run Algorithm 2 to get a subspace  $Q$ .
- 4:   If  $\dim(Q) < r$ , and there are at least  $N$  points that are within distance  $O(\sqrt{r}\epsilon/\alpha\gamma)$  add it to the collection of subspaces.
- 5: **end for**
- 6: Let  $Q$  be a subspace in the collection, remove  $Q$  if there is a subspace  $Q'$  with  $\dim(Q') < \dim(Q)$  and  $\|P_{Q^\perp} Q'\| \leq O(H\sqrt{r}\epsilon/\alpha\gamma)$
- 7: Merge all subspaces that are within distance  $O(H\sqrt{r}\epsilon/\alpha\gamma)$  to each other.

---

**Theorem 5.2.** *If  $H\sqrt{r}\epsilon/\alpha\gamma = o(\alpha)$ , then the output of Algorithm 3 contains only subspaces that are  $\epsilon_S = O(H\sqrt{r}\epsilon/\alpha\gamma)$ -close to the properly filled facets, and for every properly filled facet there is a subspace in the output that is  $\epsilon_S$  close.*

## 6. Finding intersections

Given an subset-separable NMF with  $(N, H, \gamma)$ -properly filled facets, let  $Q_i$  denote the subspace associated with a set  $S_i$  of vertices:  $Q_i = \text{span}(W^{S_i})$ . For all properly filled facets with at least two vertices, Algorithm 3 returns noisy versions of the subspaces  $\hat{Q}_i$  that are  $\epsilon_S$  close to the true subspaces. Without loss of generality, assume the first  $h$  facets are non-singletons. Our goal is to find all the intersection vertices  $\{W^i : i \in P\}$ . Recall that intersection vertices are the unique intersections of subsets of  $S_1, \dots, S_h$ . We can view this as a set intersection problem:

**Set Intersections** We are given sets  $S_1, S_2, \dots, S_h \subset [r]$ . There is an unknown set  $P \subset [r]$  such that  $\forall i \in P$  there exists  $\{S_{i_k}\}$  and  $i = \cap_k S_{i_k}$ . Our goal is to find the set  $P$ .

This problem is simple if we know the subsets of  $W^j$  in each facet. However, since what we really have access to are subspaces, it is impossible to identify the vertices unless we have a subspace of dimension 1. On the other hand, we can perform intersection and linear-span for the subspaces, which correspond to intersection and union for the sets. We also know the size of a set by looking at the dimension of the subspace. The main challenge here is that we cannot afford to enumerate all the possible combinations of the sets,

---

**Algorithm 4** Finding Intersection
 

---

**input**  $k$  sets  $S_1, \dots, S_h$ .

**output** A set  $P$  that has all the intersection vertices.

Initialize  $P = \emptyset, R = \emptyset$ .

**for**  $i = 1$  to  $r$  **do**

Let  $S = [r]$

**for**  $j = 1$  to  $h$  **do**

**if**  $|S \cap S_j| < |S|$  and  $S \cap S_j \not\subseteq R$  **then**

$S = S \cap S_j$

**end if**

**end for**

$R = R \cup S$

Add  $S$  to  $P$  if  $|S| = 1$ .

**end for**

---

and also there are vertices that are not intersection vertices and they may or may not appear in the sets we have. The idea of the algorithm is to keep vertices that we have already found in  $R$ , and try to avoid finding the same vertices by making sure  $S$  is never a subset of  $R$ . We show after every inner-loop one of the two cases can happen: in the first case we find an element in  $P$ ; in the second case  $S$  is a set that satisfies  $(S \setminus R) \cap P = \emptyset$ , so by adding  $S$  to  $R$  we remove some of the vertices that are not in  $P$ . Since the size of  $R$  increases by at least 1 in every iteration until  $R = [r]$ , the algorithm always ends in  $r$  iterations and finds all the vertices in  $P$ . In practice, we implement all the set operations in 4 using the analogous subspace operations (see Algorithm 6 in Appendix). We prove the following :

**Theorem 6.1.** *When  $W$  is  $\alpha$ -robust and  $\epsilon_S < o(\alpha^3/r^{2.5})$ , Algorithm 6 finds all the intersection vertices of  $W$ , with error at most  $\epsilon_v = 4r^{1.5}\epsilon_S/\alpha$ .*

---

**Algorithm 5** Finding remaining vertices
 

---

**input** matrix  $\hat{M}$ , intersection vertices  $\hat{W}^1, \dots, \hat{W}^{|P|}$ .

**output** remaining vertices  $\hat{W}^{|P|+1}, \dots, \hat{W}^r$ .

**for**  $i = |P| + 1$  TO  $r$  **do**

Let  $Q = \text{span}\{\hat{W}^1, \dots, \hat{W}^{i-1}\}$ .

Pick the point  $\hat{M}^j$  with largest  $\|P_{Q^\perp} \hat{M}^j\|$ , let  $\hat{W}^i = \hat{M}^j$ .

**end for**

---

**Finding the remaining vertices** The remaining vertices correspond to singleton sets in subset-separable assumption. They appear in rows of  $M$ . The situation is very similar to the separable NMF and we use an algorithm from (Arora et al., 2013) to find the remaining vertices. For completeness we describe the algorithm here. By Lemma 4.5 in (Arora et al., 2013) we directly get the following theorem:

**Theorem 6.2.** *If vertices already found have accuracy  $\epsilon_v$  such that  $\epsilon_v \leq \alpha/20r$ , Algorithm 5 outputs the remaining vertices with accuracy  $O(\epsilon/\alpha^2) < \epsilon_v$ .*

**Running time.** Face-Intersect (Algorithm 1) has 3 parts: find facets (Algorithm 3), find intersections (Algorithm 4) and find remaining anchors (Algorithm 5). We discuss the runtime of each part. We first do dimension reduction to map the  $n$  points to an  $r$ -dimensional subspace to improve the running time of later steps. The dimension reduction takes  $O(nmr)$  time, where  $n, m$  are the number of rows and columns of  $M$ , respectively, and  $r$  is the rank of the factorization. Algorithm 3's runtime is  $O(nd \cdot \text{OPT})$ , where  $d$  is the max dimension of properly filled facets (typically  $d < r \ll m$ ).  $\text{OPT}$  is the time to solve the convex optimization problem in Algorithm 2.  $\text{OPT}$  is essentially equivalent to solving an LP with  $n$  nonnegative variables and  $r + d$  constraints. Algorithm 4's runtime is  $O(kr^4)$  where  $k$  is the number of properly filled facets; typically  $k \ll n$ . Algorithm 5's runtime is  $O(nr^3)$ . The overall runtime of Face-Intersect is  $O(mnr + nd \cdot \text{OPT} + kr^4 + nr^3)$ . Calling the  $\text{OPT}$  routine is the most expensive part of the algorithm. Empirically, we find that the algorithm converges after  $\sim k \ll nd$  calls to  $\text{OPT}$ .

## 7. Generative model of NMF naturally creates properly filled facets

To better understand the generality of our approach, we analyzed a simple generative model of subset-separable NMFs and showed that properly filled facets naturally arise with high probability.

**Generative Model** Given a simplex  $W$  that is  $\alpha$ -robust and a subset of facets  $S_1, S_2, \dots, S_k$  that is subset separable. Let  $p_i$  be the probability associated with facet  $i$ , and let  $p_{\min} = \min_{i \leq k} p_i$  and  $d = \max_{i \in [k]} |S_i|$ . For convenience, denote  $S_0 = [r]$  and  $p_0 = 1 - \sum_{i=1}^k p_i$ . To generate a sample, first sample facet  $S_i$  with probability  $p_i$ , and then uniformly randomly sample a point within the convex hull of the points  $\{W^j : j \in S_i\}$ . Here we think of  $d$  as a small constant or  $O((\log n) / \log \log n)$  (in general  $d$  can be much smaller than  $r$ ). For example, separability assumption implies  $d = 1$ , and it is already nontrivial when  $d = 2$ .

**Theorem 7.1.** *Given  $n = \Omega(\max\{(4d)^d \log(d/\eta), kr^2 \log(d/p_{\min}\eta)\}/p_{\min})$  samples from the model, with high probability the facets  $S_1, \dots, S_k$  are  $(p_{\min}n/2, 200r^{1.5}/p_{\min}\alpha, \alpha^2/16d)$  properly filled.*

The proof relies on the following two lemmas. The first lemma shows that once we have enough points in a simplex, then there is a center point with high probability.

**Lemma 7.2.** *Given  $n = \Omega((4d)^d \log d/\eta)$  uniform points  $v^1, v^2, \dots, v^n$  in a standard  $d$ -dimensional simplex (with vertices  $e_1, e_2, \dots, e_d$ ), with probability  $1 - \eta$  there exists a point  $v_i$  such that  $v_i = \sum_{j \neq i} w_j v^j$  ( $w_j \geq 0, \sum_{j \neq i} w_j =$*

*1), and  $\sigma_{\min}(\sum_{j \neq i} w_j (v^j)(v^j)^T) \geq 1/16d$ .*

The next lemma shows unless a subspace contains a properly filled facet, it cannot contain too many points in its neighborhood.

**Lemma 7.3.** *Given  $n = \Omega(d^2 \log(d/p_{\min}\eta)/p_{\min})$  uniform points  $v^1, v^2, \dots, v^n$  in a standard  $d$ -dimensional simplex (with vertices  $e_1, e_2, \dots, e_d$ ), with probability  $1 - \eta$  for all matrices  $A$  whose largest column norm is equal to 1, there are at most  $p_{\min}n/4$  points with  $\|Av^i\| \leq p_{\min}/200d$ .*

## 8. Experiments

While our algorithm has strong theoretical guarantees, we additionally performed proof-of-concept experiments to show that when the noise is relatively small, our algorithm can outperform the state-of-art NMF algorithms. We simulated data according to the generative NMF model described in Section 7. We first randomly select  $r$  non-negative vectors in  $\mathbb{R}^m$  as rows of the  $W$  matrix. We grouped the vertices  $W^j$  into  $r$  groups,  $S_1, \dots, S_r$  of three elements each, such that each vertex is the unique intersection of two groups. Each  $S_i$  then corresponds to a 2-dim facet. To generate the  $A$  matrix, for each  $S_i$ , we randomly sampled  $n_1$  rows of  $A$  with support  $S_i$ , where each entry is an i.i.d. from  $\text{Unif}(0, 1)$ . An additional  $n_2$  rows of  $A$  were sampled with full support. These correspond to points in the interior of the simplex. We tested a range of settings with  $m$  between 5 to 100,  $r$  between 3 to 10, and  $n_1$  and  $n_2$  between 100 and 500. We generated the true data as  $M = AW$  and added i.i.d. Gaussian noise to each entry of  $M$  to generate the observed data  $\tilde{M}$ .

There are many algorithms for solving NMF, most of them are either iterative algorithms that have no guarantees, or algorithms that work only under separability condition. We choose two typical algorithms: the Anchor-Words algorithm (Arora et al., 2013) for separable NMF, and Projected Gradient (Lin, 2007) for iterative algorithms. For each simulated NMF, we evaluated the output factors  $\hat{A}, \hat{W}$  of these algorithms on three criteria: accuracy of the reconstructed anchors to the true anchors,  $\|W - \hat{W}\|_2$ ; accuracy of the reconstructed data matrix to the *observed* data,  $\|\tilde{M} - \hat{A}\hat{W}\|_2$ ; accuracy of the reconstructed data to the *true* data,  $\|M - \hat{A}\hat{W}\|_2$ . In Figure 2, we show the results for the three methods under the setting  $n_1 = 100, n_2 = 100, m = 10, r = 5$ . We grouped the results by the noise level of the experiment, which is defined to be the ratio of the average magnitude of the noise vectors to the average magnitude of the data points in  $\mathbb{R}^m$ . Face-Intersect is substantially more accurate in reconstructing the  $W$  matrix compared to Anchor-Words and Projected Gradient. In terms of reconstructing the  $\tilde{M}$  and  $M$  matrices, Face-Intersect slightly outperforms Anchor-Words ( $p < 0.05$  t-test), and

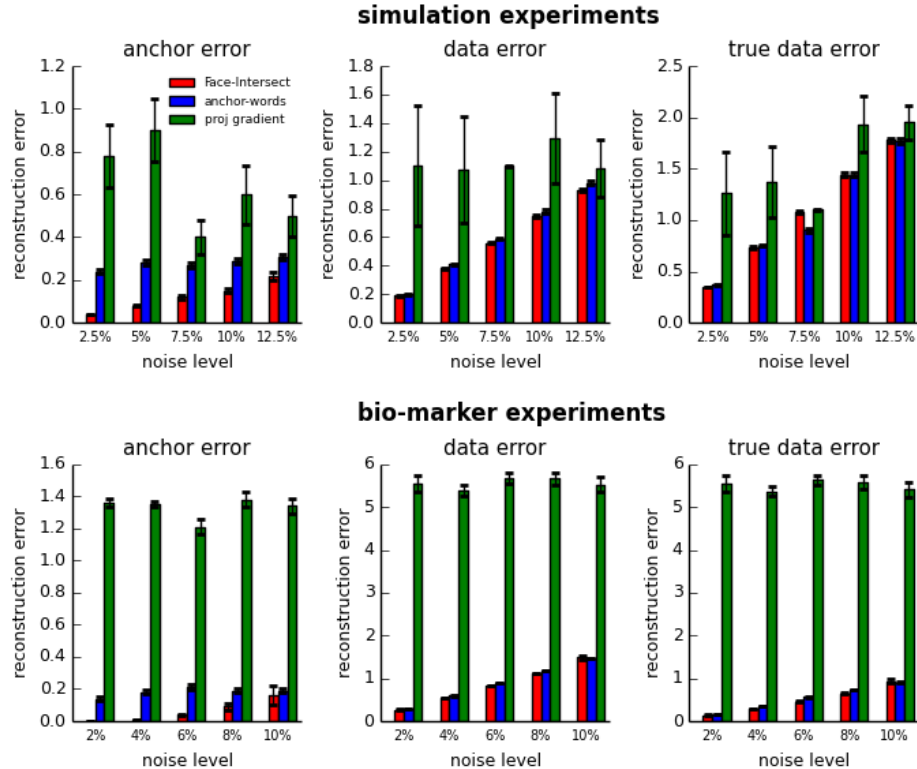


Figure 2. Reconstruction accuracy of the three NMF algorithm as a function of data noise. Standard error shown in the error bars.

they both were substantially more accurate than Projected Gradient. As noise level increased, the accuracy of Face-Intersect and Anchor-Words degrades and at noise around 12.5%, the accuracy of the three methods converged. In many applications, we are more interested in accurate reconstruction of the latent  $W$  than of  $M$ . For example, in bio-medical applications, each row of  $M$  is a sample and each column is the measurement of that sample at a particular bio-marker. Each sample is typically a mixture of  $r$  cell-types, and each cell-type corresponds to a row of  $W$ . The  $A$  matrix gives the mixture weights of the cell-types into the samples. Given measurement on a set of samples,  $M$ , an important problem is to infer the values of the latent cell-types at each bio-marker,  $W$  (Zou et al., 2014). To create a more realistic simulation of this setting, we used DNA methylation values measures at 100 markers in 5 cell-types (Monocytes, B-cells, T-cells, NK-cells and Granulocytes) as the true  $W$  matrix (Zou et al., 2014). From these 5 anchors we generated 600 samples—which is a typical size of such datasets—using the same procedure as above. Both Face-Intersect and Anchor-Words substantially outperformed Projected Gradient across all three reconstruction criteria. In terms of reconstructing the biomarker matrix  $W$ , Face-Intersect was significantly more accurate than Anchor-Words. For reconstructing the data matrices  $M$  and  $\tilde{M}$ , Face-Intersect was statistically more accurate than

Anchor-Words when the noise is less than 8% ( $p < 0.05$ ), though the magnitude of the difference is small.

**Discussion** We have presented the notion of subset separability, which substantially generalizes separable NMFs and is a necessary condition for the factorization to be unique or to have minimal volume. This naturally led us to develop the Face-Intersect algorithm, and we showed that when the NMF is subset separable and have properly filled facets, this algorithm provably recovers the true factorization. Moreover, it is robust to small adversarial noise. We show that the requirements for Face-Intersect to work are satisfied by simple generative models of NMFs. The original theoretical analysis of separable NMF led to a burst of research activity. Several highly efficient NMF algorithms were inspired by the theoretical ideas. We are hopeful that the idea of subset-separability will similarly lead to practical and theoretically sound algorithms for a much larger class of NMFs. Our Face-Intersect algorithm and its analysis is a first proof-of-concept that this is a promising direction. In exploratory experiments, we showed that under some settings where the relative noise is low, the Face-Intersect algorithm can outperform state-of-art NMF solvers. An important agenda of research will be to develop more robust and scalable algorithms motivated by our subset-separability analysis.



## References

- Arora, S., Ge, R., Kannan, R., and Moitra, A. Computing a nonnegative matrix factorization – provably. In *STOC*, pp. 145–162, 2012.
- Arora, Sanjeev, Ge, Rong, Halpern, Yoni, Mimno, David M., Moitra, Ankur, Sontag, David, Wu, Yichen, and Zhu, Michael. A practical algorithm for topic modeling with provable guarantees. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 28 (2), pp. 280–288. JMLR: W&CP, 2013.
- Bittorf, V., Recht, B., Re, C., and Tropp, J. Factoring non-negative matrices with linear programs. In *NIPS*, 2012.
- Devarajan, K. Nonnegative matrix factorization: an analytical and interpretive tool in computational biology. *PLoS Comput Biol*, 2009.
- Donoho, D. and Stodden, V. When does non-negative matrix factorization give the correct decomposition into parts? In *NIPS*, 2003.
- Elhamifar, Ehsan and Vidal, René. Sparse subspace clustering. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 2790–2797. IEEE, 2009.
- Gillis, N. Robustness analysis of hottopixx, a linear programming model for factoring nonnegative matrices. 2012. <http://arxiv.org/abs/1211.6687>.
- Gillis, N. The why and how of nonnegative matrix factorization, 2014. <http://arxiv.org/abs/1401.5226>.
- Gillis, N. and Vavasis, S.A. Fast and robust recursive algorithms for separable nonnegative matrix factorization. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(4):698–714, April 2014. ISSN 0162-8828. doi: 10.1109/TPAMI.2013.226.
- Gomez, C., Borgne, H. Le, Allemand, P., Delacourt, C., and Ledru, P. N-findr method versus independent component analysis for lithological identification in hyperspectral imagery. *Int. J. Remote Sens.*, 28(23), January 2007.
- Hardt, Moritz and Moitra, Ankur. Algorithms and hardness for robust subspace recovery. In *COLT*, pp. 354–375, 2013.
- Kumar, A., Sindhwani, V., and Kambadur, P. Fast conical hull algorithms for near-separable non-negative matrix factorization. 2012. <http://arxiv.org/abs/1210.1190v1>.
- Lee, D. and Seung, H. Learning the parts of objects by non-negative matrix factorization. *Nature*, pp. 788–791, 1999.
- Lee, D. and Seung, H. Algorithms for non-negative matrix factorization. In *NIPS*, pp. 556–562, 2000.
- Lin, Chih-Jen. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.
- Moitra, Ankur. An almost optimal algorithm for computing nonnegative rank. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6–8, 2013*, pp. 1454–1464, 2013.
- Nascimento, J.M. P. and Dias, J. M. B. Vertex component analysis: A fast algorithm to unmix hyperspectral data. *IEEE TRANS. GEOSCI. REM. SENS*, 43:898–910, 2004.
- Stewart, G.W. and Sun, J. *Matrix perturbation theory*, volume 175. Academic press New York, 1990.
- Vavasis, S. On the complexity of nonnegative matrix factorization. *SIAM Journal on Optimization*, pp. 1364–1377, 2009.
- Vidal, René. A tutorial on subspace clustering. *IEEE Signal Processing Magazine*, 28(2):52–68, 2010.
- Xu, W., Liu, X., and Gong, Y. Document clustering based on non-negative matrix factorization. In *SIGIR*, pp. 267–273, 2003.
- Zou, J., Lippert, C., Heckerman, D., Aryee, M., and Listgarten, J. Genome-wide association studies without the need for cell-type composition. *Nature Methods*, pp. 309–311, 2014.