
Approximation Algorithms for Orthogonal Non-negative Matrix Factorization

Moses Charikar
Stanford University

Lunjia Hu
Stanford University

Abstract

In the non-negative matrix factorization (NMF) problem, the input is an $m \times n$ matrix M with non-negative entries and the goal is to factorize it as $M \approx AW$. The $m \times k$ matrix A and the $k \times n$ matrix W are both constrained to have non-negative entries. This is in contrast to singular value decomposition, where the matrices A and W can have negative entries but must satisfy the orthogonality constraint: the columns of A are orthogonal and the rows of W are also orthogonal. The orthogonal non-negative matrix factorization (ONMF) problem imposes both the non-negativity and the orthogonality constraints, and previous work showed that it leads to better performances than NMF on many clustering tasks. We give the first constant-factor approximation algorithm for ONMF when one or both of A and W are subject to the orthogonality constraint. We also show an interesting connection to the correlation clustering problem on bipartite graphs. Our experiments on synthetic and real-world data show that our algorithm achieves similar or smaller errors compared to previous ONMF algorithms while ensuring perfect orthogonality (many previous algorithms do not satisfy the hard orthogonality constraint).

1 INTRODUCTION

Low-rank approximation of matrices is a fundamental technique in data analysis. Given a large data matrix M of size $m \times n$, the goal is to approximate it by a

low-rank matrix AW where A has size $m \times k$ and W has size $k \times n$. Here k is called the *inner dimension* of the factorization $M \approx AW$, controlling the rank of AW . Such low-rank matrix decomposition enables a succinct and often more interpretable representation of the original data matrix M .

One of the standard approaches of low-rank approximation is singular value decomposition (SVD) (Wold et al., 1987; Alter et al., 2000; Papadimitriou et al., 2000). SVD computes a solution minimizing both the Frobenius norm $\|M - AW\|_F$ and the spectral norm $\sigma_{\max}(M - AW)$ (Eckart and Young, 1936; Mirsky, 1960). In addition, SVD always gives a solution with the orthogonality property: the columns of A are orthogonal and the rows of W are also orthogonal. Orthogonality makes the factors more separable, and thus causes the low-rank representation to have a cleaner structure.

However, in certain cases the data matrix M is inherently non-negative, with entries corresponding to frequencies or probability mass, and in these cases SVD has a serious limitation: the factors A and W computed by SVD often contain negative entries, making the factorization less interpretable. Non-negative matrix factorization (NMF), which constrains A and W to have non-negative entries, is better suited to these cases, and is applied in many domains including computer vision (Lee and Seung, 1999; Li et al., 2001), text mining (Xu et al., 2003; Pauca et al., 2004) and bioinformatics (Brunet et al., 2004; Kim and Park, 2007; Devarajan, 2008).

One drawback of NMF relative to SVD is that it gives less separable factors: the angle between any two columns of A or any two rows of W is at most $\pi/2$ simply because the inner product of a pair of vectors with non-negative coordinates is always non-negative. To reap the benefits of non-negativity and orthogonality simultaneously, orthogonal NMF (ONMF) adds orthogonality constraints to NMF on one or both of the factors A and W : the columns of A and/or the rows of W are required to be orthogonal. Indeed, ONMF leads to better empirical performances in many clus-

tering tasks (Ding et al., 2006; Choi, 2008; Yoo and Choi, 2010). While previous works showed ONMF algorithms that converge to local minima (Ding et al., 2006) and an efficient polynomial-time approximation scheme (EPTAS) assuming the inner-dimension is a constant (Asteris et al., 2015), a theoretical understanding of the worst-case guarantee one can achieve for ONMF with arbitrary inner-dimension is lacking. In this work, we show the first constant-factor approximation algorithm for ONMF with respect to the squared Frobenius error $\|M - AW\|_F^2$ when the orthogonality constraint is imposed on one or both of the factors.

Our Results We use approximation algorithms for weighted k -means as subroutines, such as the $(9 + \varepsilon)$ -approximation local search algorithm by Kanungo et al. (2002). Assuming an r -approximation algorithm for weighted k -means, we show algorithms for ONMF with approximation ratio $2r$ in the *single-factor orthogonality* setting where only one of the factors A or W is required to be orthogonal (Theorem 3), and approximation ratio $\left(2r + \frac{8r+8}{\sin^2(\pi/12)}\right)$ in the *double-factor orthogonality* setting where both A and W are required to be orthogonal (Theorem 8). Here, A (resp. W) being orthogonal means that its columns (resp. rows) are orthogonal but not necessarily of unit length. The approximation ratios are provable upper bounds for the ratio between the error of the output (A, W) of the algorithm and the minimum error over all feasible solutions (A, W) , with error measured using the squared Frobenius norm $\|M - AW\|_F^2$. We also demonstrate the superior practical performance of our algorithms by experiments in both the single-factor and the double-factor orthogonality setting on synthetic and real-world datasets (see Section 5 and Appendix G).

Sparse Structure of Solution When we impose the orthogonality constraint on both the columns of A and the rows of W , the non-negativity and the orthogonality constraints together cause the solution to ONMF to have a very sparse structure. Let \mathbf{a}_i denote the i -th column of A and \mathbf{w}_i^\top denote the i -th row of W . Since \mathbf{a}_i and \mathbf{a}_j are constrained to have non-negative entries but zero inner product, they have disjoint supports, and this also holds for \mathbf{w}_i and \mathbf{w}_j . As a result, $AW = \sum_{i=1}^k \mathbf{a}_i \mathbf{w}_i^\top$ naturally consists of k disjoint blocks, as shown in Figure 1.

If the input M factorizes as $M = AW$ exactly, we can easily recover A and W based on the block-wise structure of M . Therefore, we focus on the agnostic setting where $M = AW$ does not hold exactly, and design approximation algorithms that find solutions

comparable to the best possible factorization.

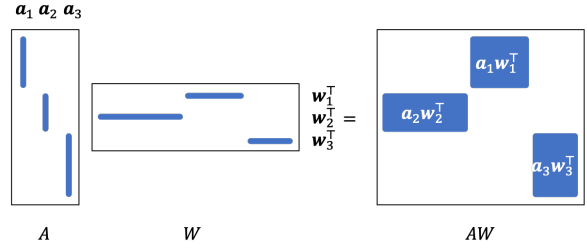


Figure 1: The k columns of A have disjoint supports. The k rows of W also have disjoint supports. The product AW has entries equal to zero outside the k blocks.

Connection to Bipartite Correlation Clustering

The block-wise structure of AW (Figure 1) relates ONMF to the correlation clustering problem (Bansal et al., 2004) on complete bipartite graphs.

To see the relationship with correlation clustering, let us consider a data matrix M with binary entries and assume $k \geq \min\{m, n\}$. Since we can find at most $\min\{m, n\}$ non-zero $\mathbf{a}_i \mathbf{w}_i^\top$ satisfying the orthogonal constraint, all $k \geq \min\{m, n\}$ give equivalent problems, where any inner-dimension is considered feasible. M can be treated as a complete bipartite graph with vertices $\{u_1, \dots, u_m\} \cup \{v_1, \dots, v_n\}$ and edges (u_i, v_j) labeled “+” if $M_{ij} = 1$ or “-” if $M_{ij} = 0$. This edge-labeled complete bipartite graph is exactly an instance of the correlation clustering problem. If the factors A and W also have binary entries and both satisfy the orthogonality constraint, the blocks of $AW = \sum_{i=1}^k \mathbf{a}_i \mathbf{w}_i^\top$ (see Figure 1) are all-ones matrices corresponding to vertex-disjoint complete bipartite sub-graphs. This is exactly the form of a solution to the correlation clustering problem, and the objective $\|M - AW\|_F^2$ is exactly the number of disagreements in the correlation clustering problem. Although our algorithm (specifically, the algorithm in Theorem 9) doesn’t impose the binary constraint on A and W , we can apply the following lemma to each block of AW to round the solution to binary with only a constant loss in the objective (see Appendix A for proof):

Lemma 1. *Let $M \in \{0, 1\}^{m \times n}$ be a binary matrix. Let $\mathbf{a} \in \mathbb{R}_{\geq 0}^m$ and $\mathbf{w} \in \mathbb{R}_{\geq 0}^n$ be two non-negative vectors. Then, there exist binary vectors $\hat{\mathbf{a}} \in \{0, 1\}^m$ and $\hat{\mathbf{w}} \in \{0, 1\}^n$ such that*

$$\|M - \hat{\mathbf{a}} \hat{\mathbf{w}}^\top\|_F^2 \leq 8 \|M - \mathbf{a} \mathbf{w}^\top\|_F^2.$$

Moreover, $\hat{\mathbf{a}}$ and $\hat{\mathbf{w}}$ can be computed in poly-time.

Thus, we can obtain an approximation algorithm for minimizing disagreements in complete bipartite graphs

via our approximation algorithm for ONMF in Theorem 9. Moreover, without the binary constraint on M, A, W , ONMF with orthogonality constraint on both A and W can be treated as a soft version of bipartite correlation clustering.

Open Questions We used the Frobenius norm as a natural measure of goodness of fit, but it would be interesting to see if one can achieve constant-factor approximation with respect to other measures, such as the spectral norm, since the two norms can be different by a factor that grows with $\min\{m, n\}$. It would also be interesting to consider replacing the orthogonality constraint on A and W by a lower bound $\theta < \pi/2$ on the angles between different columns of A and different rows of W .

Related Work Non-negative matrix factorization was first proposed by Paatero and Tapper (1994), and was shown to be NP-hard by Vavasis (2010). Algorithmic frameworks for efficiently finding local optima include the multiplicative updating framework (Lee and Seung, 2001) and the alternating non-negative least squares framework (Lin, 2007; Kim and Park, 2011). Under the usually mild *separability* assumption, Arora et al. (2016) showed an efficient algorithm that computes the global optimum.

Ding et al. (2006) first studied NMF with the orthogonality constraint, and showed its effectiveness in document clustering. After that, algorithms for ONMF using various techniques have been developed for a broad range of applications (Chen et al., 2009; Ma et al., 2010; Kuang et al., 2012; Pompili et al., 2013; Li et al., 2014b; Kim et al., 2015; Qin et al., 2016; Alaudah et al., 2017; Huang et al., 2019). The less restrictive single-factor orthogonality setting attracted the most attention, and most algorithms for solving it belong to the multiplicative updating framework: iteratively updating A and/or W by taking the element-wise product with other computed non-negative matrices (Yang and Laaksonen, 2007; Choi, 2008; Yoo and Choi, 2008, 2010; Yang and Oja, 2010; Pan and Ng, 2018; He et al., 2020). Other techniques include HALS (hierarchical alternating least squares) (Li et al., 2014a; Kimura et al., 2016) and using a penalty function (Del Buono, 2009) for the orthogonality constraint.

While improving the separability of the factors compared to NMF, these algorithm *do not guarantee convergence to a solution that has perfect orthogonality* (which is also demonstrated in our experiments). There are only a few previous algorithms that have this guarantee, including the EM-ONMF algorithm (Pompili et al., 2014), the ONMFS algorithm (Asteris et al., 2015) and the NRCG-ONMF algorithm Zhang et al.

(2016). ONMFS is the only previous algorithm we know that has a provable approximation guarantee, but it has a running time exponential in the squared inner dimension. (Pompili et al., 2014) give a reduction of ONMF to spherical k -means with a somewhat non-standard objective function: the goal is to minimize the sum of 1 minus the square of cosine similarity, while the commonly studied objective function for spherical k -means sums up 1 minus the cosine similarity. Our results for ONMF imply a constant factor approximation for this variant of spherical k -means with the squared cosine similarity in the objective. Many variants of ONMF have also been studied in the literature, including the semi-ONMF (Li et al., 2018) and the sparse ONMF (Chen et al., 2018; Li et al., 2020).

We would also like to point out that the connection between ONMF and k -means shown in (Ding et al., 2006, Theorems 1 and 2) does not give a reduction in either direction. Their proof shows that the optimization problem associated with k -means is essentially ONMF, but *with additional constraints*: the matrix G in the ONMF formulation (8) in (Ding et al., 2006) is replaced by matrix \tilde{G} in the k -means formulation (11) in (Ding et al., 2006). However \tilde{G} is a “normalized cluster indicator matrix” that is more constrained than the generic matrix G with orthonormal columns because the entries in every column of \tilde{G} are either zero or take the same non-zero value. This additional constraint makes their argument insufficient to either directly derive an algorithm for ONMF with the same approximation guarantee given one for k -means, or the other way around. Also, later works such as Yoo and Choi (2010) and Asteris et al. (2015) used techniques different from k -means to improve the empirical performance of ONMF.

The correlation clustering problem was proposed by Bansal et al. (2004) on complete graphs, who showed a constant factor approximation algorithm for the disagreement minimization version and a polynomial-time approximation scheme (PTAS) for the agreement maximization version. Ailon et al. (2008) showed a simple combinatorial algorithm achieving an approximation ratio of 3 in the disagreement minimization version, and Chawla et al. (2015) improved the approximation ratio to the currently best 2.06. Chawla et al. (2015) also showed a 3-approximation algorithm on complete k -partite graphs.

2 WEIGHTED k -MEANS

The k -means problem is a fundamental clustering problem, and we will apply algorithms for its weighted version as subroutines to solve our orthogonal NMF problem. Given points $\mathbf{m}_1, \dots, \mathbf{m}_n \in \mathbb{R}^m$ and their

weights $\ell_1, \dots, \ell_n \in \mathbb{R}_{\geq 0}$, the weighted k -means problem seeks k centroids $\mathbf{c}_1, \dots, \mathbf{c}_k$ and an assignment mapping $\phi : \{1, \dots, n\} \rightarrow \{1, \dots, k\}$ that solve the following optimization problem:

$$\underset{\mathbf{c}_1, \dots, \mathbf{c}_k; \phi}{\text{minimize}} \sum_{i=1}^n \ell_i \|\mathbf{m}_i - \mathbf{c}_{\phi(i)}\|_2^2.$$

Even the unweighted ($\forall i, \ell_i = 1$) version of this problem is APX hard, but many constant factor approximation algorithms were obtained. Kanungo et al. (2002) showed a local-search algorithm achieving an approximation ratio $9 + \varepsilon$,¹ which was improved by Ahmadian et al. (2017) in the unweighted setting to an approximation ratio 6.357.

3 SINGLE-FACTOR ORTHOGONALITY

In the single-factor orthogonality setting, we impose the orthogonality constraint only on one of the factors A or W . For concreteness, let us assume that the rows of W are required to be orthogonal. Since the rows of W are also non-negative, they must have disjoint supports, or equivalently, each column of W has at most one non-zero entry. This particular structure relates our problem closely to the weighted k -means problem, and it's not hard to apply the approximation algorithms for weighted k -means to our single-factor orthogonality setting. Specifically, assuming there is a poly-time r -approximation algorithm for weighted k -means, we show a poly-time algorithm for the single-factor orthogonality setting with approximation factor $2r$ (Theorem 3).

To see why k -means plays an important role in our problem, recall that the non-negativity and orthogonality constraints on W simplify each column \mathbf{w}_i of W to the form $\theta_i \mathbf{e}_{\phi(i)}$, where θ_i is a non-negative real number, ϕ maps $\{1, \dots, n\}$ to $\{1, \dots, k\}$, and $\mathbf{e}_{\phi(i)} \in \mathbb{R}^k$ is the unit vector with its $\phi(i)$ -th coordinate being one. This means that the i -th column of AW is exactly θ_i times the $\phi(i)$ -th column of A . If we think of the k columns of A as k centroids, and ϕ as the assignment mapping that maps every column of M to its closest centroid, (unweighted) k -means is exactly our problem with the additional constraint that $\theta_i = 1$ for all i .

With the freedom of choosing θ_i , it's more convenient to solve our problem by *weighted* k -means. Assume

¹The algorithm of Kanungo et al. (2002) was originally designed for the unweighted setting, but it works naturally in the weighted setting if we use the algorithm by Feldman et al. (2007) when computing the (k, ε) -approximate centroid set on which local search is performed.

without loss of generality that every column of A in the optimal solution is the zero vector or has unit length as we can always scale them back using θ_i . We normalize the columns of M and weight each column proportional to its initial squared L_2 norm. After that, always setting $\theta_i = 1$ only increases the approximation ratio by a factor of 2 as we show in the following lemma proved in Appendix B (think of \mathbf{x} as a column of the optimal A and \mathbf{y} as a column of M):

Fact 2. *Let $\mathbf{x} \in \mathbb{R}_{\geq 0}^m$ be a unit vector or the zero vector. For any non-negative vector $\mathbf{y} \in \mathbb{R}_{\geq 0}^m$ and any $\theta \geq 0$, we have $\|\mathbf{y} - \theta \mathbf{x}\|_2^2 \geq \frac{1}{2} \|\mathbf{y}\|_2^2 \cdot \|\bar{\mathbf{y}} - \mathbf{x}\|_2^2$, where $\bar{\mathbf{y}} = \begin{cases} \frac{\mathbf{y}}{\|\mathbf{y}\|_2}, & \mathbf{y} \neq \mathbf{0} \\ \mathbf{0}, & \mathbf{y} = \mathbf{0} \end{cases}$.*

Based on this intuition, we obtain the following algorithm. Let $\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_n \in \mathbb{R}_{\geq 0}^m$ be the columns of M , and let $\bar{\mathbf{m}}_i$ be the normalized version of \mathbf{m}_i :

$$\bar{\mathbf{m}}_i := \begin{cases} \frac{\mathbf{m}_i}{\|\mathbf{m}_i\|_2}, & \text{if } \mathbf{m}_i \neq \mathbf{0} \\ \mathbf{0}, & \text{if } \mathbf{m}_i = \mathbf{0} \end{cases}.$$

Let $\ell_i := \|\mathbf{m}_i\|_2^2$ be the *weight* of point $\bar{\mathbf{m}}_i \in \mathbb{R}_{\geq 0}^m$. We first compute an r -approximate solution to the following weighted k -means problem:

$$\underset{\mathbf{c}_1, \dots, \mathbf{c}_k; \phi}{\text{minimize}} \sum_{i=1}^n \ell_i \|\bar{\mathbf{m}}_i - \mathbf{c}_{\phi(i)}\|_2^2. \quad (1)$$

We can assume WLOG that all of the centroids $\mathbf{c}_1, \dots, \mathbf{c}_k$ have non-negative coordinates since increasing the negative coordinates to zero never increases the weighted k -means objective. Then we simply output $A = [\mathbf{c}_1, \dots, \mathbf{c}_k]$ and $W = [\theta_1 \mathbf{e}_{\phi(1)}, \dots, \theta_n \mathbf{e}_{\phi(n)}]$, where

$$\theta_i = \begin{cases} \frac{\langle \mathbf{m}_i, \mathbf{c}_{\phi(i)} \rangle}{\|\mathbf{c}_{\phi(i)}\|_2^2}, & \text{if } \mathbf{c}_{\phi(i)} \neq \mathbf{0} \\ 0, & \text{if } \mathbf{c}_{\phi(i)} = \mathbf{0} \end{cases} \in \arg \min_{\theta} \|\mathbf{m}_i - \theta \mathbf{c}_{\phi(i)}\|_2^2.$$

We show the approximation guarantee in the following theorem proved in Appendix C.

Theorem 3. *The algorithm above computes a $2r$ -approximate solution A and W in the single-factor orthogonality setting in time $O(T_{k\text{-means}} + mn)$, where $T_{k\text{-means}}$ is the time needed by the weighted k -means subroutine.*

4 DOUBLE-FACTOR ORTHOGONALITY

Now we consider the double-factor orthogonality setting, where we require A to have orthogonal columns and W to have orthogonal rows, and show a poly-time constant factor approximation algorithm in this setting.

We first state some basic facts that will be used in the discussion of our algorithms.

Useful Inequalities The following *doubled triangle inequality* for the squared L_2 distance between vectors \mathbf{x} and \mathbf{y} is useful when we analyze the approximation ratio of our algorithm:

Fact 4. $\|\mathbf{x} - \mathbf{y}\|_2^2 \leq 2\|\mathbf{x}\|_2^2 + 2\|\mathbf{y}\|_2^2$.

When both \mathbf{x} and \mathbf{y} have non-negative coordinates, we have the following stronger fact:

Fact 5. *If both \mathbf{x} and \mathbf{y} have non-negative coordinates, then $\|\mathbf{x} - \mathbf{y}\|_2^2 \leq \|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2$.*

Center of Mass Given n points $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^m$ and their weights $\ell_1, \dots, \ell_n \in \mathbb{R}_{\geq 0}$, the point $\mathbf{y} \in \mathbb{R}^m$ minimizing the weighted sum of the squared L_2 distances $\sum_{i=1}^n \ell_i \|\mathbf{x}_i - \mathbf{y}\|_2^2$ is the center of mass: $\mathbf{y} = (\sum_{i=1}^n \ell_i \mathbf{x}_i) / (\sum_{i=1}^n \ell_i)$. Moreover, the weighted sum can be decomposed using the following identity (see, for example, Lemma 2.1 in (Kanungo et al., 2002)):

Fact 6. *Assume $\ell_1, \dots, \ell_n \geq 0$ and $\mathbf{y} = (\sum_{i=1}^n \ell_i \mathbf{x}_i) / (\sum_{i=1}^n \ell_i)$. Then for any vector \mathbf{b} , we have*

$$\sum_{i=1}^n \ell_i \|\mathbf{x}_i - \mathbf{b}\|_2^2 = \sum_{i=1}^n \ell_i \|\mathbf{x}_i - \mathbf{y}\|_2^2 + \sum_{i=1}^n \ell_i \|\mathbf{y} - \mathbf{b}\|_2^2.$$

4.1 Intuition

We describe the intuition that leads us to the algorithm. As our first step, we solve the weighted k -means problem as we did in the single-factor orthogonality setting, but we need to additionally ensure that the columns of A are orthogonal. By the *doubled triangle inequality* (Fact 4) and the property of the center of mass (Fact 6), we can move the n points to their centroids without affecting the approximation ratio too much. Now there are only k distinct points, and it's more convenient to treat these points as vectors, so that we can talk about the angles between them. Our goal is to find k *orthogonal* centroids that approximate these k vectors. The key challenge is to find the assignment mapping: which vectors are mapped to the same centroid, and once we know the assignment mapping, we can find the best centroids by optimizing each coordinate separately (see (2)). Intuitively, the assignment mapping should respect the angles between the vectors: if a pair of vectors form a “small” angle, they should be mapped to the same centroid, and if they form a “large” angle close to $\pi/2$, they should be mapped to different centroids. However, two vectors both forming “small” angles with the third may themselves form a relatively “large” angle. In order to solve the lack of transitivity, we need to eliminate

angles that are neither very “small” nor very “large”. We make the observation that if the angle between two vectors is in the range $[\pi/6, \pi/3]$, they can't be simultaneously close to a set of orthonormal vectors, and thus they can't have low cost in the optimal solution, so we can safely “ignore” them by decreasing their weights by the same amount. This *weight reduction* procedure eventually makes the angle between any two vectors lie in the range $[0, \pi/6) \cup (\pi/3, \pi/2]$. If two vectors both have angles less than $\pi/6$ with the third, they themselves cannot form an angle larger than $\pi/3$, so now we have the desired transitivity. Our Lemma 10 shows that the assignment mapping computed this way is comparable to the optimal one.

4.2 Algorithm

Our algorithm consists of three major steps. The first step is to apply the weighted k -means algorithm as we did in the single-factor orthogonality setting, and two additional steps are needed to make sure the solution has both factors being orthogonal.

Step 1: Weighted k -Means

Let $\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_n \in \mathbb{R}_{\geq 0}^m$ be the columns of M and define $\bar{\mathbf{m}}_i$ and ℓ_i the same way as in Section 3. Compute an r -approximate solution $\mathbf{c}_1, \dots, \mathbf{c}_k, \phi$ to the weighted k -means problem (1). Define the weight q_j of a centroid \mathbf{c}_j to be the total weight of the points assigned to it: $q_j := \sum_{i \in \phi^{-1}(j)} \ell_i$. By Fact 6, we can always assume WLOG that whenever $q_j > 0$, it holds that $\mathbf{c}_j = (\sum_{i \in \phi^{-1}(j)} \ell_i \bar{\mathbf{m}}_i) / q_j$. Under this assumption, whenever $q_j > 0$, we have $\|\mathbf{c}_j\|_2 \leq 1$. We also have the following easy fact:

Fact 7. *If $q_j > 0$, then $\mathbf{c}_j \neq \mathbf{0}$.*

Proof. Assume for the sake of contradiction that $\mathbf{c}_j = \mathbf{0}$. According to our assumption, we have $\mathbf{0} = \mathbf{c}_j = (\sum_{i \in \phi^{-1}(j)} \ell_i \bar{\mathbf{m}}_i) / q_j$, so for all $i \in \phi^{-1}(j)$, $\ell_i \bar{\mathbf{m}}_i = \mathbf{0}$. If $\bar{\mathbf{m}}_i \neq \mathbf{0}$, we know $\ell_i = 0$; otherwise, we know $\mathbf{m}_i = \mathbf{0}$ and thus, again, $\ell_i = \|\mathbf{m}_i\|_2^2 = 0$. Now we have our desired contradiction: $q_j = \sum_{i \in \phi^{-1}(j)} \ell_i = 0$. \square

Step 2: Weight Reduction

Recall that the weight q_j of a centroid \mathbf{c}_j was defined to be the total weight of the points assigned to it. The second step of the algorithm is to *reduce* the weights q_1, \dots, q_k to q'_1, \dots, q'_k . To start, all q'_j are initialized to be q_j . Our algorithm iterates over all pairs (j_1, j_2) satisfying $1 \leq j_1 < j_2 \leq k$. If $q'_{j_1} > 0, q'_{j_2} > 0$ and $\angle(\mathbf{c}_{j_1}, \mathbf{c}_{j_2}) \in [\pi/6, \pi/3]$, our algorithm decreases both q'_{j_1}, q'_{j_2} by the minimum of the two (thus sending at least one of them to 0). Recall Fact 7 that \mathbf{c}_{j_1} and

\mathbf{c}_{j_2} are both non-zero, so the angle between them is well-defined.

Step 3: Finalize the Solution

Now we are most interested in centroids \mathbf{c}_j with positive weights ($q'_j > 0$) after the weight reduction step. For any two centroids $\mathbf{c}_{j_1}, \mathbf{c}_{j_2}$ with positive weights, we know $\angle(\mathbf{c}_{j_1}, \mathbf{c}_{j_2}) \in [0, \pi/6) \cup (\pi/3, \pi/2]$. Since the angles between vectors satisfy the triangle inequality, we can group these centroids so that $\angle(\mathbf{c}_{j_1}, \mathbf{c}_{j_2}) \in [0, \pi/6)$ if j_1, j_2 belong to the same group, and $\angle(\mathbf{c}_{j_1}, \mathbf{c}_{j_2}) \in (\pi/3, \pi/2]$ if j_1, j_2 belong to different groups. Suppose \mathbf{c}_j belongs to group $\sigma(j) \in \{1, \dots, k\}$.

We claim that we can find an *optimal* solution to the following optimization problem in poly-time:

$$\begin{aligned} & \underset{\mathbf{a}_1, \dots, \mathbf{a}_k}{\text{minimize}} \quad \sum_{j: q'_j > 0} q'_j \|\mathbf{c}_j - \mathbf{a}_{\sigma(j)}\|_2^2, \\ & \text{s.t. } \mathbf{a}_1, \dots, \mathbf{a}_k \in \mathbb{R}_{\geq 0}^m, \\ & \quad \forall 1 \leq s < t \leq k, \mathbf{a}_s^\top \mathbf{a}_t = 0. \end{aligned} \quad (2)$$

To solve the above optimization problem, we decompose it *coordinate-wise*. Specifically, the constraints on $\mathbf{a}_1, \dots, \mathbf{a}_k$ can be translated to that for every $h \in \{1, \dots, m\}$, the h -th coordinates $a_{1,h}, \dots, a_{k,h}$ are all non-negative and contain at most one positive value. The objective can also be decomposed coordinate-wise:

$$\begin{aligned} \sum_{j: q'_j > 0} q'_j \|\mathbf{c}_j - \mathbf{a}_{\sigma(j)}\|_2^2 &= \sum_{h=1}^m \sum_{j: q'_j > 0} q'_j (c_{j,h} - a_{\sigma(j),h})^2 \\ &\stackrel{\text{def}}{=} \sum_{h=1}^m O_h. \end{aligned}$$

If we define q_s^* as the total weight of the s -th group: $q_s^* = \sum_{j \in \sigma^{-1}(s)} q'_j$, and when $q_s^* > 0$ define $\mu_{s,h}$ as the weighted average of the h -th coordinate of the centroids in the s -th group: $\mu_{s,h} = (q_s^*)^{-1} \sum_{j \in \sigma^{-1}(s)} q'_j c_{j,h}$, we can further decompose the objective above using Fact 6 as

$$O_h = \sum_{j: q'_j > 0} q'_j (c_{j,h} - \mu_{\sigma(j),h})^2 + \sum_{s: q_s^* > 0} q_s^* (\mu_{s,h} - a_{s,h})^2.$$

The first term does not depend on $\mathbf{a}_1, \dots, \mathbf{a}_k$, and the second term is minimized when $a_{s,h} = \mu_{s,h}$ for $s = \arg \max_s q_s^* \mu_{s,h}^2$ and $a_{s,h} = 0$ for other s . We have thus computed the optimal solution to (2). Since $\|\mathbf{c}_j\|_2 \leq 1$ whenever $q'_j > 0$, it is straightforward to check that $\|\mathbf{a}_s\|_2 \leq 1$ for $s = 1, \dots, k$.

We output $A = [\mathbf{a}_1, \dots, \mathbf{a}_k]$ and $W = [\theta_1 \mathbf{e}_{\sigma(\phi(1))}, \dots, \theta_n \mathbf{e}_{\sigma(\phi(n))}]$ as the final solution,

where

$$\begin{aligned} \theta_i &= \begin{cases} \frac{\langle \mathbf{m}_i, \mathbf{a}_{\sigma(\phi(i))} \rangle}{\|\mathbf{a}_{\sigma(\phi(i))}\|_2^2}, & \text{if } \mathbf{a}_{\sigma(\phi(i))} \neq \mathbf{0} \\ 0, & \text{if } \mathbf{a}_{\sigma(\phi(i))} = \mathbf{0} \end{cases} \\ &\in \arg \min_{\theta} \|\mathbf{m}_i - \theta \mathbf{a}_{\sigma(\phi(i))}\|_2^2. \end{aligned}$$

Note that $\sigma(j)$ was defined only for j with $q'_j > 0$, but here we extend its definition to all $j \in \{1, \dots, k\}$ by setting the remaining values arbitrarily.

4.3 Analysis

We show the following two theorems on the approximation guarantee of our algorithm in the double-factor orthogonality setting. Theorem 8 applies to general inner dimensions k , while Theorem 9 gives improved approximation factors when k is large, which is the case when we apply our ONMF algorithm to correlation clustering. Recall that we used an r -approximation algorithm for weighted k -means as a subroutine, and we assume that its running time is $T_{k\text{-means}}$.

Theorem 8. *The algorithm in Section 4.2 computes a $\left(2r + \frac{8r+8}{\sin^2(\pi/12)}\right)$ -approximate solution A and W in the double-factor orthogonality setting in time $O(T_{k\text{-means}} + mn + mk^2)$.*

Theorem 9. *When $k \geq \min\{m, n\}$, there exists an algorithm that gives a $\frac{1}{\sin^2(\pi/12)} (\leq 15)$ -approximate solution in the double-factor orthogonality setting in time $O(mn^2)$.*

We prove Theorem 8 based on the following lemma, which we prove in Appendix D. We defer the proof of Theorem 9 to Appendix E.

Lemma 10. *Let $\mathbf{z}_1, \dots, \mathbf{z}_{k_1} \in \mathbb{R}_{\geq 0}^m$ be non-negative unit vectors that are orthogonal to each other. For any $\sigma' : \{1, \dots, k\} \rightarrow \{1, \dots, k_1\}$, we have*

$$\sum_{j=1}^k q_j \|\mathbf{c}_j - \mathbf{a}_{\sigma(j)}\|_2^2 \leq \frac{2}{\sin^2(\pi/12)} \sum_{j=1}^k q_j \|\mathbf{c}_j - \mathbf{z}_{\sigma'(j)}\|_2^2.$$

Proof of Theorem 8. We obtain the running time of the algorithm by summing over the three steps. Step 1 requires $O(mn)$ time to create the input to the weighted k -means subroutine, and the subroutine takes $T_{k\text{-means}}$ time. Step 2 takes $O(mk^2)$ time because we use $O(m)$ time to compute the angle between each of the $O(k^2)$ pairs of centroids. In step 3, it takes $O(mk)$ time to solve the optimization problem (2), and it takes time $O(mn)$ to compute the θ_i 's.

The feasibility of (A, W) is clear from the algorithm. We focus on proving the approximation guarantee. We start by showing an upper bound for the objective $\|M - AW\|_F^2$ achieved by our algorithm. For

$i = 1, \dots, n$, the i -th column of AW is $\theta_i \mathbf{a}_{\sigma(\phi(i))}$, where $\theta_i \in \arg \min_{\theta} \|\mathbf{m}_i - \theta \mathbf{a}_{\sigma(\phi(i))}\|_2^2$. Therefore,

$$\begin{aligned} & \|M - AW\|_F^2 \\ &= \sum_{i=1}^n \|\mathbf{m}_i - \theta_i \mathbf{a}_{\sigma(\phi(i))}\|_2^2 \\ &\leq \sum_{i=1}^n \|\mathbf{m}_i - \|\mathbf{m}_i\|_2 \mathbf{a}_{\sigma(\phi(i))}\|_2^2 \\ &= \sum_{i=1}^n \|\mathbf{m}_i\|_2^2 \cdot \|\bar{\mathbf{m}}_i - \mathbf{a}_{\sigma(\phi(i))}\|_2^2. \end{aligned}$$

By Fact 6 and $\mathbf{c}_j = \left(\sum_{i \in \phi^{-1}(j)} \ell_i \bar{\mathbf{m}}_i\right) / q_j$, we have

$$\begin{aligned} & \|M - AW\|_F^2 \\ &\leq \sum_{i=1}^n \|\mathbf{m}_i\|_2^2 \cdot \|\bar{\mathbf{m}}_i - \mathbf{a}_{\sigma(\phi(i))}\|_2^2 \\ &= \sum_{i=1}^n \ell_i \cdot \|\bar{\mathbf{m}}_i - \mathbf{a}_{\sigma(\phi(i))}\|_2^2 \\ &= \sum_{i=1}^n \ell_i \cdot \|\bar{\mathbf{m}}_i - \mathbf{c}_{\phi(i)}\|_2^2 + \sum_{i=1}^n \ell_i \cdot \|\mathbf{c}_{\phi(i)} - \mathbf{a}_{\sigma(\phi(i))}\|_2^2 \\ &= \sum_{i=1}^n \ell_i \cdot \|\bar{\mathbf{m}}_i - \mathbf{c}_{\phi(i)}\|_2^2 + \sum_{j=1}^k q_j \|\mathbf{c}_j - \mathbf{a}_{\sigma(j)}\|_2^2. \quad (3) \end{aligned}$$

(3) gives an upper bound for $\|M - AW\|_F^2$. We proceed by giving a lower bound for the objective $\|M - A^{\text{opt}}W^{\text{opt}}\|_F^2$ achieved by the optimal solution $(A^{\text{opt}}, W^{\text{opt}})$. We first remove the columns of A^{opt} filled with the zero vector and also remove the corresponding rows in W^{opt} . This doesn't change the product $A^{\text{opt}}W^{\text{opt}}$ and doesn't violate the orthogonality requirement either, but the sizes of A^{opt} and W^{opt} may now change to $m \times k_1$ and $k_1 \times n$. We can now assume WLOG that every column $\mathbf{a}_s^{\text{opt}}$ of A^{opt} is a unit vector. Note that each column of W contains at most one non-zero entry, so we have

$$\begin{aligned} & \|M - A^{\text{opt}}W^{\text{opt}}\|_F^2 \\ &\geq \sum_{i=1}^n \min_{\substack{1 \leq s \leq k_1 \\ \theta \geq 0}} \|\mathbf{m}_i - \theta \mathbf{a}_s^{\text{opt}}\|_2^2 \\ &\geq \frac{1}{2} \sum_{i=1}^n \|\mathbf{m}_i\|_2^2 \cdot \min_{1 \leq s \leq k_1} \|\bar{\mathbf{m}}_i - \mathbf{a}_s^{\text{opt}}\|_2^2 \\ &= \frac{1}{2} \sum_{i=1}^n \ell_i \cdot \min_{1 \leq s \leq k_1} \|\bar{\mathbf{m}}_i - \mathbf{a}_s^{\text{opt}}\|_2^2, \quad (4) \end{aligned}$$

where the second inequality follows from Fact 2. By the r -approximate optimality of $\mathbf{c}_1, \dots, \mathbf{c}_k$, we have

$$\|M - A^{\text{opt}}W^{\text{opt}}\|_F^2$$

$$\begin{aligned} &\geq \frac{1}{2} \sum_{i=1}^n \ell_i \cdot \min_{1 \leq s \leq k_1} \|\bar{\mathbf{m}}_i - \mathbf{a}_s^{\text{opt}}\|_2^2 \\ &\geq \frac{1}{2r} \sum_{i=1}^n \ell_i \cdot \|\bar{\mathbf{m}}_i - \mathbf{c}_{\phi(i)}\|_2^2. \quad (5) \end{aligned}$$

Combining (4) with (5), we have

$$\begin{aligned} & (4r + 4) \|M - A^{\text{opt}}W^{\text{opt}}\|_F^2 \\ &\geq \sum_{i=1}^n \ell_i (2 \|\bar{\mathbf{m}}_i - \mathbf{c}_{\phi(i)}\|_2^2 + 2 \min_{1 \leq s \leq k_1} \|\bar{\mathbf{m}}_i - \mathbf{a}_s^{\text{opt}}\|_2^2) \\ &\geq \sum_{i=1}^n \ell_i \min_{1 \leq s \leq k_1} \|\mathbf{c}_{\phi(i)} - \mathbf{a}_s^{\text{opt}}\|_2^2 \\ &= \sum_{i=1}^n \ell_i \|\mathbf{c}_{\phi(i)} - \mathbf{a}_{\sigma'(\phi(i))}^{\text{opt}}\|_2^2 \\ &= \sum_{j=1}^k q_j \|\mathbf{c}_j - \mathbf{a}_{\sigma'(j)}^{\text{opt}}\|_2^2, \quad (6) \end{aligned}$$

where (6) is by Fact 4 and $\sigma'(j)$ is defined to be $\arg \min_{1 \leq s \leq k_1} \|\mathbf{c}_j - \mathbf{a}_s^{\text{opt}}\|_2$. Applying Lemma 10, we get

$$\begin{aligned} & (4r + 4) \|M - A^{\text{opt}}W^{\text{opt}}\|_F^2 \\ &\geq \sum_{j=1}^k q_j \|\mathbf{c}_j - \mathbf{a}_{\sigma'(j)}^{\text{opt}}\|_2^2 \quad (7) \end{aligned}$$

$$\geq \frac{\sin^2(\pi/12)}{2} \sum_{j=1}^k q_j \|\mathbf{c}_j - \mathbf{a}_{\sigma(j)}\|_2^2. \quad (8)$$

Combining (3) with (5) and (7), we have

$$\begin{aligned} & \|M - AW\|_F^2 \\ &\leq \sum_{i=1}^n \ell_i \cdot \|\bar{\mathbf{m}}_i - \mathbf{c}_{\phi(i)}\|_2^2 + \sum_{j=1}^k q_j \|\mathbf{c}_j - \mathbf{a}_{\sigma(j)}\|_2^2 \\ &\leq \left(2r + \frac{8r + 8}{\sin^2(\pi/12)}\right) \|M - A^{\text{opt}}W^{\text{opt}}\|_F^2. \end{aligned}$$

□

5 EXPERIMENTS

We report on the results of experiments comparing the performance of our algorithm with eight previous algorithms in the literature. For these experiments, we use k -means++ as the subroutine for solving k -means. For the single factor orthogonality setting, our experiments show that our algorithm ensures perfect orthogonality and give similar approximation error as six previous algorithms in the literature that do not guarantee orthogonality. For this single factor setting, we also directly compare to two previous algorithms that

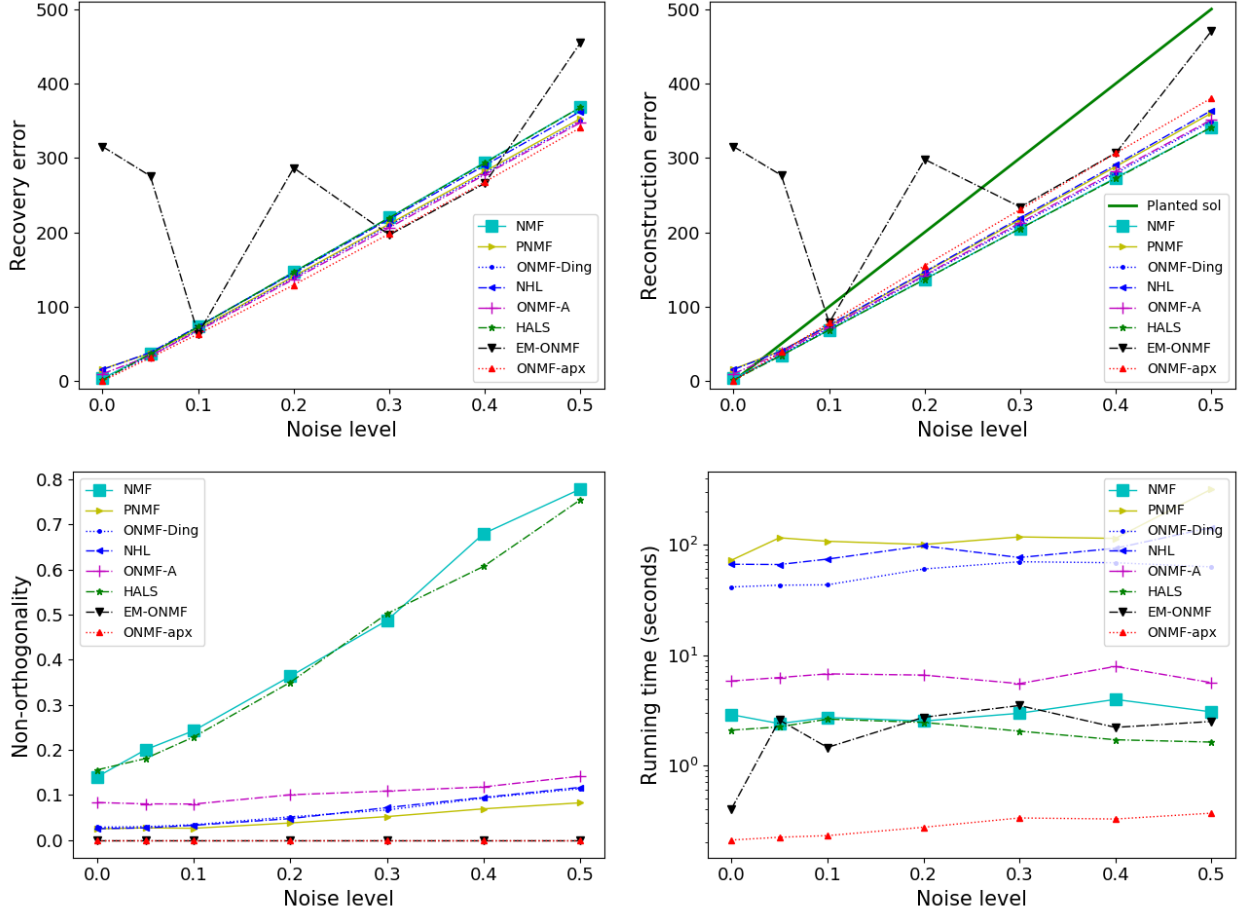


Figure 2: Results of experiment 1. From left to right, the plots in the first row show the recovery error and the reconstruction error, and the plots in the second row show the non-orthogonality and the running time. The performance of our algorithm is shown in the red line under the label ONMF-apx.

do ensure orthogonality and find that that the performance of our algorithm is superior. One of the previous algorithms has runtime that scales very poorly with inner dimension (and worse error for small inner dimension); the other suffers from poor local minima, leading to large error even with zero noise. For the double factor orthogonality setting, only two previous algorithms are able to handle this case. None of them ensure perfect orthogonality, while our algorithm does. Further, it has lower error than these previous algorithms. Our algorithm runs significantly faster than all these other algorithms in both settings. Thus we achieve the best of both worlds – stronger approximation guarantees as well as superior practical performance for ONMF.

Specifically, we compare our algorithm (ONMF-apx) with previous algorithms in the more well-studied single-factor orthogonality setting on synthetic data, and defer the experiments on real-world data and in the double-factor orthogonality setting to Appendix

G. The previous algorithms we compare with include NMF (Lee and Seung, 2001), PNMf (Yuan and Oja, 2005), ONFS-Ding (Ding et al., 2006), NHL (Yang and Laaksonen, 2007), ONMF-A (Choi, 2008), HALS (Li et al., 2014a), EM-ONMF (Pompili et al., 2014), and ONMFS (Asteris et al., 2015).

Experimental Setup We generate the input matrix $M \in \mathbb{R}^{m \times n}$ by adding noise to the product M_{truth} of random non-negative matrices $A_{\text{truth}} \in \mathbb{R}^{m \times k}$ and $W_{\text{truth}} \in \mathbb{R}^{k \times n}$. We make sure that W_{truth} has orthogonal rows², and every non-zero entry of A_{truth} and W_{truth} is independently drawn from the exponential distribution with mean 1. We call $M_{\text{truth}} = A_{\text{truth}}W_{\text{truth}}$ the *planted solution*, and we add iid noise to every entry of M_{truth} to obtain M . The noise also

²Due to non-negativity, making the rows of W_{truth} orthogonal is equivalent to making every column of W_{truth} contain at most one non-zero entry. Independently for every column, we pick the location of the non-zero entry uniformly at random.

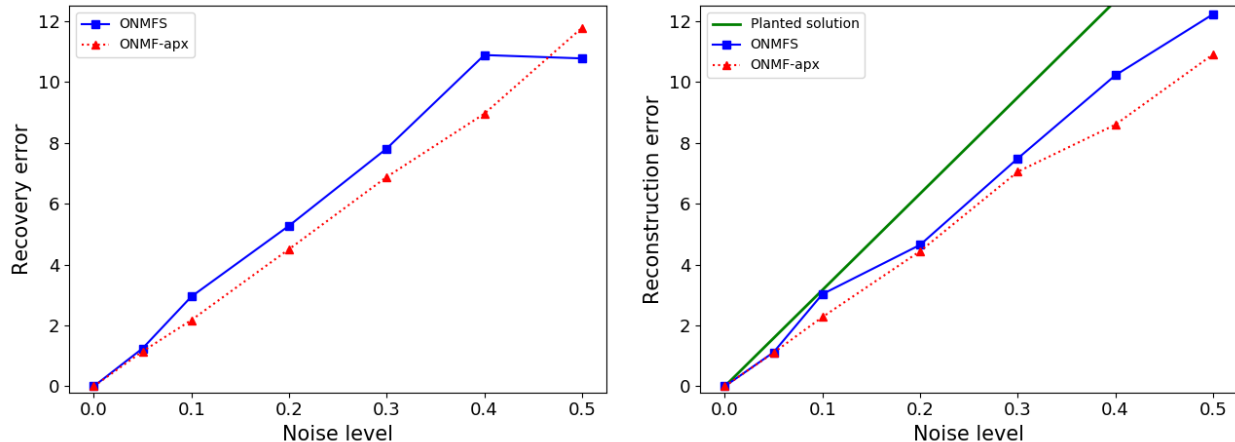


Figure 3: Results of experiment 2. From left to right, the plots show the recovery error and the reconstruction error. The non-orthogonality (not shown in figure) is identically zero for both algorithms.

follows an exponential distribution, and we use the phrase “noise level” to denote the mean of that distribution.

Evaluation We measure the quality of the matrices A and W output by the algorithms in terms of the approximation error and the orthogonality of W . We measure the approximation error using the Frobenius norm: we compute both the *recovery error* $\|M_{\text{truth}} - AW\|_F$, which measures how well the output recovers the underlying structure of the input, and the *reconstruction error* $\|M - AW\|_F$, which measures the approximation error to the input matrix that contains iid noise. We define the reconstruction error of the planted solution M_{truth} as $\|M - M_{\text{truth}}\|_F$, whose value concentrates well around $\sqrt{2mn}$ times the noise level as shown in the following easy fact:

Fact 11. *The mean (resp. standard deviation) of $\|M - M_{\text{truth}}\|_F^2$ is $2mn$ (resp. $\sqrt{20mn}$) times the noise level squared.*

We measure the non-orthogonality of W by the Frobenius norm of $WW^T - I$ after removing the zero rows of W and normalizing the other rows.

Experiment 1 In the first experiment, we choose $m = 100, n = 5000, k = 10$, and compare our algorithm with previous ones. We run each algorithm independently for 7 times and record the median results in Figure 2. We found that ONMFS could not finish in a reasonable amount of time, so we investigate it separately on smaller matrices in experiment 2. We also found that there is a high variance in the approximation error of EM-ONMF because it often converges to a bad local optimum, giving the fluctuating black lines in Figure 2.

As shown in Figure 2, our algorithm ensures perfect orthogonality and gives similar approximation error as previous ones which do not guarantee orthogonality. Except for EM-ONMF, none of the other previous algorithms in this experiment output a perfectly orthogonal W . Our recovery error is slightly better than previous algorithms, but our reconstruction error is slightly worse. This is because the orthogonality constraint effectively regularizes our solution, making it fit the noise in the input worse but reveal the structure of the input better. It is worth noting that our algorithm achieves lower reconstruction errors than the planted solution M_{truth} , and so do most other algorithms in the experiment (the reconstruction error of M_{truth} concentrates well around 1000 times the noise level (thick green line in Figure 2) by Fact 11).

We would also like to point out that our algorithm runs significantly faster than all the other algorithms considered in this experiment. The bottom right plot of Figure 2 shows the running time on a machine with 1.4 GHz Quad-Core Intel Core i5 processor and 8 GB 2133 MHz LPDDR3 memory (note that the y -axis is on logarithmic scale). Our algorithm is based on the k -means++ subroutine, which is very efficient. The previous algorithms are based on iterative update and may take a long time to reach a local optimum.

Experiment 2 We compare our algorithm with ONMFS (Asteris et al., 2015), an algorithm that guarantees perfect orthogonality, but runs in time exponential in the squared inner dimension. ONMFS was based on two levels of exhaustive search, which is inefficient when the inner dimension is large. We thus reduce the sizes of the matrices and set $m = 10, n = 50, k = 2$ in this experiment. Our result shows that our algorithm gives smaller error than ONMFS (Figure 3).

Acknowledgements

We thank Suyash Gupta and anonymous reviewers for helpful comments on earlier versions of this paper.

Moses Charikar was supported by a Simons Investigator Award, a Google Faculty Research Award and an Amazon Research Award. Lunjia Hu was supported by NSF Award IIS-1908774 and a VMware fellowship.

References

- Sara Ahmadian, Ashkan Norouzi-Fard, Ola Svensson, and Justin Ward. Better guarantees for k-means and Euclidean k-median by primal-dual algorithms. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 61–72. IEEE, 2017.
- Nir Ailon, Moses Charikar, and Alantha Newman. Aggregating inconsistent information: ranking and clustering. *Journal of the ACM (JACM)*, 55(5):1–27, 2008.
- YK Alaudah, Haibin Di, and Ghassan AlRegib. Weakly supervised seismic structure labeling via orthogonal non-negative matrix factorization. In *79th EAGE Conference and Exhibition 2017*, volume 2017, pages 1–5. European Association of Geoscientists & Engineers, 2017.
- Orly Alter, Patrick O Brown, and David Botstein. Singular value decomposition for genome-wide expression data processing and modeling. *Proceedings of the National Academy of Sciences*, 97(18):10101–10106, 2000.
- Sanjeev Arora, Rong Ge, Ravi Kannan, and Ankur Moitra. Computing a nonnegative matrix factorization—provably. *SIAM Journal on Computing*, 45(4):1582–1611, 2016.
- Megasthenis Asteris, Dimitris Papailiopoulos, and Alexandros G Dimakis. Orthogonal NMF through subspace exploration. In *Advances in Neural Information Processing Systems*, pages 343–351, 2015.
- Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Machine learning*, 56(1-3):89–113, 2004.
- Jean-Philippe Brunet, Pablo Tamayo, Todd R Golub, and Jill P Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the national academy of sciences*, 101(12):4164–4169, 2004.
- Shuchi Chawla, Konstantin Makarychev, Tselil Schramm, and Grigory Yaroslavtsev. Near optimal LP rounding algorithm for correlation clustering on complete and complete k-partite graphs. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 219–228, 2015.
- Gang Chen, Fei Wang, and Changshui Zhang. Collaborative filtering using orthogonal nonnegative matrix tri-factorization. *Information Processing & Management*, 45(3):368–379, 2009.
- Yong Chen, Hui Zhang, Rui Liu, and Zhiwen Ye. Soft orthogonal non-negative matrix factorization with sparse representation: Static and dynamic. *Neuro-computing*, 310:148–164, 2018.
- Seungjin Choi. Algorithms for orthogonal nonnegative matrix factorization. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pages 1828–1832. IEEE, 2008.
- Nicoletta Del Buono. A penalty function for computing orthogonal non-negative matrix factorizations. In *2009 Ninth International Conference on Intelligent Systems Design and Applications*, pages 1001–1005. IEEE, 2009.
- Karthik Devarajan. Nonnegative matrix factorization: an analytical and interpretive tool in computational biology. *PLoS computational biology*, 4(7), 2008.
- Chris Ding, Tao Li, Wei Peng, and Haesun Park. Orthogonal nonnegative matrix t-factorizations for clustering. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 126–135, 2006.
- Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL <http://archive.ics.uci.edu/ml>.
- Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.
- Dan Feldman, Morteza Monemizadeh, and Christian Sohler. A PTAS for k-means clustering based on weak coresets. In *Proceedings of the twenty-third annual symposium on Computational geometry*, pages 11–18, 2007.
- Ping He, Xiaohua Xu, Jie Ding, and Baichuan Fan. Low-rank nonnegative matrix factorization on Stiefel manifold. *Information Sciences*, 514:131–148, 2020.
- Meng Huang, JiHong OuYang, Chen Wu, and Liu Bo. Collaborative filtering based on orthogonal non-negative matrix factorization. In *Journal of Physics: Conference Series*, volume 1345, page 052062. IOP Publishing, 2019.
- Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. A local search approximation algorithm for k-means clustering. In *Proceedings of the eighteenth annual symposium on Computational geometry*, pages 10–18, 2002.

- Hyunsoo Kim and Haesun Park. Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics*, 23(12):1495–1502, 2007.
- Jingu Kim and Haesun Park. Fast nonnegative matrix factorization: An active-set-like method and comparisons. *SIAM Journal on Scientific Computing*, 33(6):3261–3281, 2011.
- Sungchul Kim, Lee Sael, and Hwanjo Yu. A mutation profile for top-k patient search exploiting geneontology and orthogonal non-negative matrix factorization. *Bioinformatics*, 31(22):3653–3659, 2015.
- Keigo Kimura, Mineichi Kudo, and Yuzuru Tanaka. A column-wise update algorithm for nonnegative matrix factorization in Bregman divergence with an orthogonal constraint. *Machine learning*, 103(2):285–306, 2016.
- Da Kuang, Chris Ding, and Haesun Park. Symmetric nonnegative matrix factorization for graph clustering. In *Proceedings of the 2012 SIAM international conference on data mining*, pages 106–117. SIAM, 2012.
- Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.
- Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.
- Bo Li, Guoxu Zhou, and Andrzej Cichocki. Two efficient algorithms for approximately orthogonal non-negative matrix factorization. *IEEE Signal Processing Letters*, 22(7):843–846, 2014a.
- Jack Yutong Li, Ruoqing Zhu, Annie Qu, Han Ye, and Zhankun Sun. Semi-orthogonal non-negative matrix factorization. *arXiv preprint arXiv:1805.02306*, 2018.
- Ping Li, Jiajun Bu, Yi Yang, Rongrong Ji, Chun Chen, and Deng Cai. Discriminative orthogonal nonnegative matrix factorization with flexibility for data representation. *Expert systems with applications*, 41(4):1283–1293, 2014b.
- Stan Z Li, Xin Wen Hou, Hong Jiang Zhang, and Qian Sheng Cheng. Learning spatially localized, parts-based representation. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.
- Wenbo Li, Jicheng Li, Xuenian Liu, and Liqiang Dong. Two fast vector-wise update algorithms for orthogonal nonnegative matrix factorization with sparsity constraint. *Journal of Computational and Applied Mathematics*, 375:112785, 2020.
- Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.
- Huifang Ma, Weizhong Zhao, Qing Tan, and Zhongzhi Shi. Orthogonal nonnegative matrix tri-factorization for semi-supervised document co-clustering. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 189–200. Springer, 2010.
- L. Mirsky. Symmetric gauge functions and unitarily invariant norms. *Quart. J. Math. Oxford Ser. (2)*, 11: 50–59, 1960. ISSN 0033-5606. doi: 10.1093/qmath/11.1.50. URL <https://doi.org/10.1093/qmath/11.1.50>.
- Pentti Paatero and Unto Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.
- Junjun Pan and Michael K Ng. Orthogonal nonnegative matrix factorization by sparsity and nuclear norm optimization. *SIAM Journal on Matrix Analysis and Applications*, 39(2):856–875, 2018.
- Christos H Papadimitriou, Prabhakar Raghavan, Hisao Tamaki, and Santosh Vempala. Latent semantic indexing: A probabilistic analysis. *Journal of Computer and System Sciences*, 61(2):217–235, 2000.
- V Paul Pauca, Fariyal Shahnaz, Michael W Berry, and Robert J Plemmons. Text mining using non-negative matrix factorizations. In *Proceedings of the 2004 SIAM International Conference on Data Mining*, pages 452–456. SIAM, 2004.
- Filippo Pompili, Nicolas Gillis, François Glineur, and Pierre-Antoine Absil. Onp-mf: An orthogonal non-negative matrix factorization algorithm with application to clustering. In *ESANN*. Citeseer, 2013.
- Filippo Pompili, Nicolas Gillis, P-A Absil, and François Glineur. Two algorithms for orthogonal nonnegative matrix factorization with application to clustering. *Neurocomputing*, 141:15–25, 2014.
- Yaoyao Qin, Caiyan Jia, and Yafang Li. Community detection using nonnegative matrix factorization with orthogonal constraint. In *2016 Eighth International Conference on Advanced Computational Intelligence (ICACI)*, pages 49–54. IEEE, 2016.
- Stephen A Vavasis. On the complexity of nonnegative matrix factorization. *SIAM Journal on Optimization*, 20(3):1364–1377, 2010.
- SVante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

- Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 267–273, 2003.
- Zhirong Yang and Jorma Laaksonen. Multiplicative updates for non-negative projections. *Neurocomputing*, 71(1-3):363–373, 2007.
- Zhirong Yang and Erkki Oja. Linear and nonlinear projective nonnegative matrix factorization. *IEEE Transactions on Neural Networks*, 21(5):734–749, 2010.
- Ji-Ho Yoo and Seung-Jin Choi. Nonnegative matrix factorization with orthogonality constraints. *Journal of computing science and engineering*, 4(2):97–109, 2010.
- Jiho Yoo and Seungjin Choi. Orthogonal nonnegative matrix factorization: Multiplicative updates on Stiefel manifolds. In *International conference on intelligent data engineering and automated learning*, pages 140–147. Springer, 2008.
- Zhijian Yuan and Erkki Oja. Projective nonnegative matrix factorization for image compression and feature extraction. In *Scandinavian Conference on Image Analysis*, pages 333–342. Springer, 2005.
- Wei Emma Zhang, Mingkui Tan, Quan Z Sheng, Lina Yao, and Qingfeng Shi. Efficient orthogonal non-negative matrix factorization over Stiefel manifold. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1743–1752, 2016.