Factor-Bounded Nonnegative Matrix Factorization

KAI LIU, Clemson University
XIANGYU LI, Colorado School of Mines
ZHIHUI ZHU, University of Denver
LODEWIJK BRAND and HUA WANG, Colorado School of Mines

Nonnegative Matrix Factorization (NMF) is broadly used to determine class membership in a variety of clustering applications. From movie recommendations and image clustering to visual feature extractions, NMF has applications to solve a large number of knowledge discovery and data mining problems. Traditional optimization methods, such as the Multiplicative Updating Algorithm (MUA), solves the NMF problem by utilizing an auxiliary function to ensure that the objective monotonically decreases. Although the objective in MUA converges, there exists no proof to show that the learned matrix factors converge as well. Without this rigorous analysis, the clustering performance and stability of the NMF algorithms cannot be guaranteed. To address this knowledge gap, in this article, we study the factor-bounded NMF problem and provide a solution algorithm with proven convergence by rigorous mathematical analysis, which ensures that both the objective and matrix factors converge. In addition, we show the relationship between MUA and our solution followed by an analysis of the convergence of MUA. Experiments on both toy data and real-world datasets validate the correctness of our proposed method and its utility as an effective clustering algorithm.

CCS Concepts: • **Computing methodologies** → **Machine learning algorithms**;

Additional Key Words and Phrases: Nonnegative matrix factorization, global sequence convergence, alternating minimization, factor boundedness constraint

ACM Reference format:
Kai Liu, Xiangyu Li, Zhihui Zhu, Lodewijk Brand, and Hua Wang. 2021. Factor-Bounded Nonnegative Matrix Factorization. *ACM Trans. Knowl. Discov. Data* 15, 6, Article 111 (May 2021), 18 pages.
https://doi.org/10.1145/3451395

1 INTRODUCTION

Nonnegative Matrix Factorization (NMF) aims at finding two nonnegative matrices, $F \in \mathbb{R}_+^{m\times r}$ and $G \in \mathbb{R}_+^{r\times n}$, whose product can well approximate an input nonnegative data matrix $X \in \mathbb{R}_+^{m\times n}$, i.e., $X \approx FG$. In general, one can interpret the columns of X as data points and the rows of X as observations (features). A broadly used objective to learn NMF is to minimize the following

The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.
This work was supported in part by the National Science Foundation (NSF) under the grants of IIS 1652943, IIS 1849359, CNS 1932482, and CCF 2029543.
Authors' addresses: K. Liu, Clemson University, Clemson, SC 29634; email: liukaizhijia@gmail.com; X. Li, L. Brand, and H. Wang (corresponding authors), Colorado School of Mines, Golden, CO 80401; emails: {lixiangyu, lbrand}@mymail.mines.edu, huawangcs@gmail.com; Z. Zhu, University of Denver, Denver, CO, 80208; email: zhihuizhu90@gmail.com.

1556-4681/2021/05-ART111 $15.00
https://doi.org/10.1145/3451395

objective [20]:

$$\min_{F,G\geq 0} \; h(F,G) = \frac{1}{2}||X - FG||_F^2, \tag{1}$$

where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. If we consider the r columns of F the basis vectors, every column of G approximates the corresponding data point in X by a linear combination of these r bases vectors, where the elements of the column of G specify the coefficients to compute the linear combination. NMF has been found useful in a large variety of real-world applications, such as image feature extraction [19], document clustering [36], single speech separation [29], music transcription [31], and protein–protein interaction network analysis [33], to name a few.

To solve the NMF objective in Equation (1), the **Multiplicative Updating Algorithm** (**MUA**) was derived using the following updating rules [19, 20]:

$$G_{ij} \leftarrow G_{ij}\frac{(F^TX)_{ij}}{(F^TFG)_{ij}}, \quad F_{ij} \leftarrow F_{ij}\frac{(XG^T)_{ij}}{(FGG^T)_{ij}}. \tag{2}$$

The convergence of this algorithm was proven using the auxiliary function method [19, 20], whose correctness was also analyzed in [8, 9, 10].

Using the optimization framework in Equation (2), many machine learning methods have been developed from NMF in the past two decades. For example, to promote the robustness against outlying features and data points, not-squared ℓ_2-norm distances were used in the NMF objectives in [11, 17, 23]; to perform clustering in the low-dimensional subspace, the interrelations between data and features were incorporated in the NMF framework in [6, 21, 35]; to leverage the local consistency, manifold regularized NMF objectives were studied in [4, 12, 13]. Despite their successes, these new methods suffered from several critical drawbacks because of MUA, such as easily being trapped into suboptimal local minima [22, 38] and not-unique solutions due to soft labeling [10, 34]. To address these shortcomings, several recent studies have proposed to solve NMF using other optimization methods. For example, the alternating nonnegative least-squares method was used in [7] to optimize the objective by solving the least-squares problems of F and G one at a time, while the other variable is fixed. In [27, 30, 39], it was proposed to decouple F and G into columns and rows and to update each row or column separately, one at a time when others are fixed. Among others, gradient approaches have been studied to improve various aspects of the solution to NMF [7, 14]. To be more specific, the gradients of function $h(F,G)$ with respect to F and G are computed as

$$\begin{aligned} \nabla_F h(F,G) &= (FG - X)G^T, \\ \nabla_G h(F,G) &= F^T(FG - X). \end{aligned} \tag{3}$$

A quick glance at Equation (2) and Equation (3) shows that they are very similar in that the former performs element-wise updating, while the latter performs matrix-wise computation. In the gradient descent method, F and G are updated by computing

$$\begin{aligned} F_{ij} &= F_{ij} - \lambda\nabla_F h(F,G)_{ij} = F_{ij} - \lambda[(FG - X)G^T]_{ij}, \\ G_{ij} &= G_{ij} - \mu\nabla_G h(F,G)_{ij} = G_{ij} - \mu[F^T(FG - X)]_{ij}. \end{aligned} \tag{4}$$

It can be verified that when $\lambda = \frac{F_{ij}}{(FGG^T)_{ij}}$ and $\mu = \frac{G_{ij}}{(F^TFG)_{ij}}$, Equation (4) is identical to Equation (2), which is close to traditional gradient descent method except for (1) element-wise update, and (2) varied learning rates during iterations. On the other hand, while the updating rules in Equation (2) can nicely guarantee the nonnegativity of both factor matrices F and G during

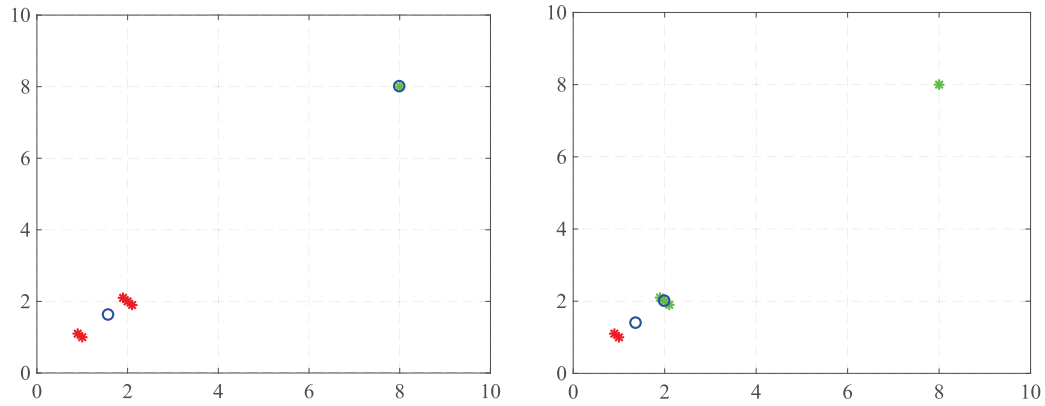

Fig. 1. Clustering with data outliers. Two data points lie around (1, 1) and three others lie around (2, 2), with one additional outlying data point lying around (8,8) that plays the role of outliers. **Left:** clustering result with vanilla MUA. **Right:** clustering result when element-wise bounds ($1 \leq F_{ij} \leq 2$) are constrained on NMF. The blue circles denote the learned centroids of the two clusters and the green and red colors denote cluster membership of the input data. Obviously, the clustering result shown in the right panel makes more sense for most real-world applications.

updating, Equation (4) fails to achieve the same goal. To tackle this, a projected gradient descent method [7] was used to update F and G using the following projections:

$$\begin{aligned} F &= \max(F - \lambda \nabla_F h(F, G), 0), \\ G &= \max(G - \lambda \nabla_G h(F, G), 0), \end{aligned} \tag{5}$$

where λ is the step size. However, the implementation details of this updating algorithm were not provided in [7] and the convergence properties of this updating algorithm were not analyzed in [7].

A recent work [22, 32, 40] constrained the two factor matrices of NMF with both lower and upper bounds. Although the solution algorithm was provided in [22], its convergence analysis was referred to [5], which, though, can only guarantee the subsequence convergence of F and G, separately. Moreover, this work presumed that the optimization problem is convex, which is apparently not true. More recently, the **block coordinate descent** (**BCD**) method (also known as the Gauss–Seidel method) was proposed and used in convergence analyses [15, 28, 37]. However, in these works, either there exists no convergence proof or only subsequence convergence can be guaranteed. In a word, no proof yet can show the sequence convergence of the solution algorithms of NMF with bounding constraints on factor matrices, although the factor bounds in NMF are very important from the following two perspectives: (1) interpretability in real-world applications, for example, the numerical values of the elements of the feature matrix F should be ranged from 0 to 255 for image analyses; otherwise, it is difficult to interpret their meanings; and (2) potential ability to tolerate outliers in clustering tasks, as illustrated in Figure 1.

With the above recognitions, in this article, we study the NMF problem with the following constraints to bound the factor matrices:

$$0 \leq f_l \leq F_{ij} \leq f_u, \quad 0 \leq g_l \leq G_{ij} \leq g_u, \tag{6}$$

which leads to an apparently much more challenging, yet more meaningful and more useful, optimization problem. To solve this optimization problem, in this article, we derive an algorithm that satisfies the constraints in Equation (6) with rigorously proven mathematical properties: (1) the objective function decreases monotonically (same as the MUA method); and (2) F and G converge

as well (***not*** guaranteed by the MUA method). Moreover, our new method can be further developed to solve the NMF problems with the energy (norm) constraints on factor matrices, where the same convergence properties can be guaranteed by rigorous mathematical proofs.

2 FACTOR-BOUNDED NMF

When we apply NMF to document clustering [36] or image clustering [19], the factors F and G can be, respectively, interpreted as the feature matrix and the membership matrix [8, 10]. With this interpretation, it is expected that the elements in F and G are within some prescribed ranges, rather than being arbitrary or extremely large. However, such constraints are not easily satisfied in conventional NMF problems due to the following reason. It can verified that if (F, G) is a solution to the NMF problem in Equation (1), then $(F' = FA, G' = A^{-1}G)$ is also a solution to the same objective, where A can be any positive diagonal matrix. This renders the ambiguity issue that we may find a solution to Equation (1) in which the diagonal elements of A are very large, such that F' has dominated entries compared with G' that lead to significantly deteriorated learning performance.

To overcome the above ambiguity issue, given the data matrix $X \in \mathbb{R}^{m\times n}$, we attempt to factorize it into the feature matrix F and the membership matrix G under certain conditions:

$$\min_{F,G} \; h(F,G) = \|X - FG\|_F^2, \quad s.t.\ F \in \mathbb{F}, G \in \mathbb{G}. \tag{7}$$

Here, we first consider the element-wise constraints:

$$\begin{aligned} \mathbb{F} &= \mathbb{F}_1 := \{F : F_L \le F \le F_U\}, \\ \mathbb{G} &= \mathbb{G}_1 := \{G : G_L \le G \le G_U\}, \end{aligned} \tag{8}$$

where $F_L, F_U \in \mathbb{R}^{m\times r}$ and $G_L, G_U \in \mathbb{R}^{r\times n}$ are prescribed parameters, and $A \ge B$ means that every element in A is greater than or equal to the corresponding element in B. Obviously, the constraints in Equation (8) can avoid the unstable solutions when the values of F or G are very large. Moreover, these constraints can also be used to incorporate prior information of an application when the ranges of the factors F and G are known in advance. Here we note that, when we choose $F_L = G_L = 0$ and F_U and F_G to be large enough, Equation (7) reduces to the conventional NMF objective in Equation (1).

2.1 Alternating Projected Gradient Descent Algorithm

Let $\mathcal{P}_{\mathbb{F}}$ be the orthogonal projector onto $\mathbb{F}$, *i.e.*,

$$\left(\mathcal{P}_{\mathbb{F}}(F)\right)(i,j) = \begin{cases} F_L(i,j), & F(i,j) < F_L(i,j), \\ F_U(i,j), & F(i,j) > F_U(i,j), \\ F(i,j), & \text{otherwise.} \end{cases} \tag{9}$$

Here $A(i,j) = A_{ij}$ represents the (i,j)th element of A. Similar notation holds for $\mathcal{P}_{\mathbb{G}}$.

We utilize the alternating minimization approach for solving Equation (7). Accordingly, we have

$$\begin{aligned} F_k &= \mathcal{P}_{\mathbb{F}}\left(F_k - \lambda \nabla_F h(F_{k-1}, G_{k-1})\right) \\ &= \mathcal{P}_{\mathbb{F}}\left(F_k - \lambda \nabla_F Tr\left[(X - F_{k-1}G_{k-1})^T (X - F_{k-1}G_{k-1})\right]\right) \\ &= \mathcal{P}_{\mathbb{F}}\left(F_k - \lambda\, (F_{k-1}G_{k-1}G_{k-1}^T - XG_{k-1}^T)\right) \end{aligned} \tag{10}$$

and

$$\begin{aligned} G_k &= \mathcal{P}_{\mathbb{G}}\Big(G_{k-1} - \mu\nabla_G h(F_k, G_{k-1})\Big) \\ &= \mathcal{P}_{\mathbb{G}}\Big(G_{k-1} - \mu\nabla_G Tr\Big[(X - F_k G_{k-1})^T (X - F_k G_{k-1})\Big]\Big) \\ &= \mathcal{P}_{\mathbb{G}}\Big(G_{k-1} - \mu\ (F_k^T F_k G_{k-1} - F_k^T X)\Big). \end{aligned} \tag{11}$$

ALGORITHM 1: Alternating Projected Gradient Descent (PGrad) for the Problem in Equation (7).

Input: data $X \in \mathbb{R}^{m\times n}$, rank of factors r, step sizes λ, μ, sets $\mathbb{F}, \mathbb{G}$, number of iterations K.
Initialization: $F_0 \in \mathbb{R}^{m\times r}, G_0 \in \mathbb{R}^{r\times n}$.
for $k = 1$ **to** K **do**
$F_k = \mathcal{P}_{\mathbb{F}}\Big(F_k - \lambda\ (F_{k-1}G_{k-1}G_{k-1}^T - XG_{k-1}^T)\Big)$,
$G_k = \mathcal{P}_{\mathbb{G}}\Big(G_{k-1} - \mu\ (F_k^T F_k G_{k-1} - F_k^T X)\Big)$,
end for
Output: F_K and G_K.

Remark 1. The step sizes λ and μ are fixed through the iterations for simplicity of the following convergence analysis in Section 3, but we note that they can be varied in each iteration to speed up the convergence. In particular, in the kth iteration, one can choose $\lambda_k \in (\underline{\lambda}, \min(\overline{\lambda}, \frac{1}{L_k}))$, where L_k satisfies the following Lipschitz condition:

$$\forall F, F' \quad \Big\|\nabla h(F, G_{k-1}) - \nabla h(F', G_{k-1})\Big\|_F \le L_k \Big\|F - F'\Big\|_F. \tag{12}$$

Then, we can obtain an upper bound for L_k as follows:

$$L_k = \lambda_{\max}\Big(\nabla^2 h(F, G_{k-1})\Big) = \Big\|G_{k-1}G_{k-1}^T\Big\|_2. \tag{13}$$

Thus, one can choose $\lambda_k \in (\underline{\lambda}, \min(\overline{\lambda}, \frac{1}{\|G_{k-1}G_{k-1}^T\|_2}))$ (and similar for μ), where $\underline{\lambda}$ and $\overline{\lambda}$ are prescribed parameters. One may also utilize other more sophisticated methods, such as the backtracking line search method, to locate the optimal step sizes of μ and λ.

Remark 2. When we set $\mathbb{F} = \{F \in \mathbb{R}^{m\times r} : F \ge 0\}$ and $\mathbb{G} = \{G \in \mathbb{R}^{r\times n} : G \ge 0\}$, Algorithm 1 almost reduces to Equation (5), which was proposed in [7]. Equation (11) differs from Equation (5) in that once F_k is updated, as can be seen in Equation (11), G_k is updated with F_k (asynchronized update) rather than the previous one F_{k-1} (synchronized update). This slight difference can speed up the convergence of Algorithm 1, based on which we will give sequence convergence proof.

3 CONVERGENCE ANALYSIS

In this section, we prove the convergence of Algorithm 1, which solves the objective in Equation (7) with constraints on $\mathbb{F}$ and $\mathbb{G}$, as defined in Equation (8).

To begin, we first show that $h(F, G)$ has a Lipschitz continuous gradient at $F \in \mathbb{F}$ and $G \in \mathbb{G}$.

Proposition 1. *The objective $h(F, G)$ has a Lipschitz continuous gradient at $F \in \mathbb{F}$ and $G \in \mathbb{G}$, where $\mathbb{F}$ and $\mathbb{G}$ are defined in Equation (8). That is, there exists a constant L_c such that*

$$\Big\|\nabla h(F, G) - \nabla h(F', G')\Big\|_F \le L_c \Big\|(F, G) - (F', G')\Big\|_F, \tag{14}$$

for all $F, F' \in \mathbb{F}$ and $G, G' \in \mathbb{F}$. Here $L_c > 0$ is referred to as the Lipschitz constant.

Proof of Proposition 1. It is equivalent to show $\|\nabla^2 h(F, G)\|_2 \le L_c$ for all $F \in \mathbb{F}, G \in \mathbb{G}$. Standard computations give the Hessian quadrature form $[\nabla^2 h(F, G)](\Delta, \Delta)$ for any $\Delta = \begin{bmatrix}\Delta_F \\ \Delta_G^{\mathrm{T}}\end{bmatrix} \in$

$\mathbb{R}^{(n+m)\times r}$ (where $\Delta_F \in \mathbb{R}^{m\times r}$ and $\Delta_G \in \mathbb{R}^{r\times n}$) as

$$\left[\nabla^2 h(F,G)\right](\Delta,\Delta) = \|\Delta_F G + F\Delta_G\|_F^2 + 2\langle FG - X, \Delta_F\Delta_G\rangle, \tag{15}$$

which gives that

$$\begin{aligned}\left\|\nabla^2 h(F,G)\right\|_2 &= \max_{\|\Delta\|_F=1} \left|\left[\nabla^2 h(F,G)\right](\Delta,\Delta)\right| \\ &\le \max_{\|\Delta\|_F=1} \|\Delta_F G + F\Delta_G\|_F^2 + 2\left|\langle FG - X, \Delta_F\Delta_G\rangle\right| \\ &\le 2\left(\|F_U\|_F^2 + \|G_U\|_F^2 + \|F_U\|_F\|G_U\|_F + \|X\|_F\right) = L_c,\end{aligned} \tag{16}$$

where the second inequality follows from $|\langle A,B\rangle| \le \|A\|_F\|B\|_F$ and the third inequality utilizes $\|CD\|_F \le \|C\|_F\|D\|_F$. This completes the proof of Proposition 1. □

To analyse the convergence, we rewrite Equation (7) as

$$\min_{F,G} f(F,G) = h(F,G) + \delta_{\mathbb{F}}(F) + \delta_{\mathbb{G}}(G), \tag{17}$$

where $\delta_{\mathbb{F}}(F) = \begin{cases} 0, & F \in \mathbb{F} \\ \infty, & F \notin \mathbb{F} \end{cases}$ is the indicator function of the set $\mathbb{F}$. The following result establishes that the subsequence convergence property of the proposed algorithm, *i.e.*, the sequence generated by Algorithm 1 is bounded and any of its limit point is a critical point of Equation (17).

Theorem 1 (Subsequence convergence). *Let $\{W_k\}_{k\ge0} = \{(F_k, G_k)\}_{k\ge0}$ be the sequence generated by Algorithm 1 with a constant step size $\lambda, \mu < \frac{1}{L_c}$. Then, the sequence $\{W_k\}_{k\ge0}$ is bounded and obeys the following properties:*

(P1) sufficient decrease:

$$f(W_{k-1}) - f(W_k) \ge \frac{\frac{1}{\max(\lambda,\mu)} - L_c}{2}\|W_k - W_{k-1}\|_F^2, \tag{18}$$

which implies that

$$\lim_{k\to\infty} \|W^{k-1} - W^k\|_F = 0; \tag{19}$$

(P2) the sequence $\{f(W_k)\}_{k\ge0}$ is convergent;

(P3) for any convergent subsequence $\{W_{k'}\}$, its limit point $W^\star$ is a critial point of f and

$$\lim_{k'\to\infty} f(W_{k'}) = \lim_{k\to\infty} f(W_k) = f(W^\star). \tag{20}$$

Proof of Theorem 1 (P1): First note that for all k, by the definition of Equation (11), we always have $\delta_{\mathbb{F}}(F_k) = \delta_{\mathbb{G}}(G_k) = 0$ and thus $f(W_k) = h(W_k)$.

Since $h(F,G)$ has a Lipschitz continuous gradient at $F \in \mathbb{F}, G \in \mathbb{G}$ with Lipschitz gradient L_c and $\frac{1}{\lambda} > L_c$, we define $h_{L_c}(F,F',G)$ as proximal regularization of $h(F,G)$ linearized at F',G:

$$\underbrace{h(F',G) + \left\langle \nabla_F h(F',G), F - F'\right\rangle + \frac{L_c}{2}\|F - F'\|_F^2}_{h_{L_c}(F,F',G)},$$

which yields the following relationship:

$$h(F,G) \le h_{L_c}(F,F',G). \tag{21}$$

Now we note that

$$\begin{aligned} F_k &= \mathcal{P}_{\mathbb{F}}\left(F_{k-1} - \lambda \nabla_F h(F_{k-1}, G_{k-1})\right) \\ &= \arg\min_{F\in\mathbb{F}} \left\|F - \left(F_{k-1} - \lambda \nabla_F h(F_{k-1}, G_{k-1})\right)\right\|_F^2 \\ &= \arg\min_{F\in\mathbb{F}} h(F_{k-1}, G_{k-1}) + \frac{1}{2\lambda}\|F - F_{k-1}\|_F^2 + \left\langle \nabla_F h(F_{k-1}, G_{k-1}), F - F_{k-1}\right\rangle \\ &= \arg\min_{F\in\mathbb{F}} h_\lambda(F, F_{k-1}, G_{k-1}), \end{aligned} \tag{22}$$

which implies that

$$h_\lambda(F_k, F_{k-1}, G_{k-1}) \le h(F_{k-1}, G_{k-1}). \tag{23}$$

Combining Equation (21) and Equation (23), we have

$$\begin{aligned} &h(F_{k-1}, G_{k-1}) - h(F_k, G_{k-1}) \\ &\ge h_\lambda(F_k, F_{k-1}, G_{k-1}) - h_{L_c}(F_k, F_{k-1}, G_{k-1}) \\ &= \frac{\frac{1}{\lambda} - L_c}{2}\|F_k - F_{k-1}\|_F^2. \end{aligned} \tag{24}$$

Similarly, we have

$$h(F_k, G_{k-1}) - h(F_k, G_k) \ge \frac{\frac{1}{\mu} - L_c}{2}\|G_k - G_{k-1}\|_F^2, \tag{25}$$

which together with the above equation gives Equation (18). Now repeating Equation (18) for all k, we get

$$\left(\frac{1}{\max(\lambda,\mu)} - L_c\right)\sum_{k=1}^{\infty}\|W_k - W_{k-1}\|_F^2 \le f(W_0), \tag{26}$$

which gives Equation (19).

(P2) It follows from Equation (18) that $\{f(W_k)\}_{k\ge 0}$ is a decreasing sequence. Due to the fact that f is lower bounded as $f(W_k) \ge 0$ for all k, we conclude that $\{f(W_k)\}_{k\ge 0}$ is convergent.

(P3) Since $F_{k'} \in \mathbb{F}, G_{k'} \in \mathbb{G}$ for all k' and both of the sets $\mathbb{F}$ and $\mathbb{G}$ are closed, we have $F^\star \in \mathbb{F}, G^\star \in \mathbb{G}$. Since h is continuous, we have

$$\lim_{k'\to\infty} f(W_{k'}) = \lim_{k'\to\infty} h(F_{k'}, G_{k'}) + \delta_{\mathbb{F}}(F_{k'}) + \delta_{\mathbb{G}}(G_{k'}) = f(W^\star), \tag{27}$$

which together with the fact that $\{f(W_k)\}_{k\ge 0}$ is convergent gives Equation (20). To show $W^\star$ is a critical point, we first rewrite Equation (22) as

$$F_k = \arg\min h_\lambda(F, F_{k-1}, G_{k-1}) + \delta_{\mathbb{F}}(F). \tag{28}$$

The optimality condition gives

$$-\nabla_F h(F_{k-1}, G_{k-1}) - \frac{1}{\lambda}(F_k - F_{k-1}) \in \partial\delta_{\mathbb{F}}(F_k). \tag{29}$$

Similarly, we have

$$-\nabla_G h(F_k, G_{k-1}) - \frac{1}{\mu}(G_k - G_{k-1}) \in \partial\delta_{\mathbb{F}}(G_k), \tag{30}$$

Now, we define

$$\underbrace{\nabla_F h(F_k, G_k) - \nabla_F h(F_{k-1}, G_{k-1}) - \frac{1}{\lambda}(F_k - F_{k-1})}_{A_k}, \tag{31}$$

$$\underbrace{\nabla_G h(F_k, G_k) - \nabla_G h(F_k, G_{k-1}) - \frac{1}{\mu}(G_k - G_{k-1})}_{B_k}, \tag{32}$$

by which we have

$$A_k \in \partial_F f(F_k, G_k), \quad B_k \in \partial_G f(F_k, G_k). \tag{33}$$

It follows from the above that

$$\begin{aligned} \lim_{k\to\infty} \|A_k\|_F &\le \lim_{k\to\infty} \left\|\nabla_F h(F_k, G_k) - \nabla_F h(F_{k-1}, G_{k-1})\right\|_F + \frac{1}{\lambda}\|F_k - F_{k-1}\|_F \\ &\le \lim_{k\to\infty} \left(L_c + \frac{1}{\lambda}\right)\|W_k - W_{k-1}\| = 0. \end{aligned} \tag{34}$$

With similar argument, we have

$$\lim_{k\to\infty} \|B_k\|_F \le \lim_{k\to\infty} \left(L_c + \frac{1}{\mu}\right)\|W_k - W_{k-1}\| = 0. \tag{35}$$

Now summing over Equation (34)–(35), we have

$$dist\Big(0, \partial f(W_k)\Big) \le \left(2L_c + \frac{1}{\lambda} + \frac{1}{\mu}\right)\|W_k - W_{k-1}\|. \tag{36}$$

Owing to the closedness properties of $\partial f(W_{k'})$ and (19), we finally obtain $0 \in \partial f(W^\star)$. Thus, $W^\star$ is a critical point of f. This completes the proof of Theorem 1. □

Theorem 2 (Sequence convergence). *The sequence $\{W_k\}_{k\ge 0}$ generated by Algorithm 1 with a constant step size $\lambda, \mu < \frac{1}{L_c}$ is global-sequence convergence.*

Proof of Theorem 2. Before proving Theorem 2, we give out an important definition.

Definition 1 (Kurdyka–Lojasiewicz Property) [3]. We say a proper semi-continuous function $\rho(\boldsymbol{u})$ satisfies **Kurdyka–Lojasiewicz (KL)** property, if $\overline{\boldsymbol{u}}$ is a critical point of $\rho(\boldsymbol{u})$, then there exist $\delta > 0$, $\theta \in [0, 1)$, $C_1 > 0$, *s.t.*

$$|\rho(\boldsymbol{u}) - \rho(\overline{\boldsymbol{u}})|^\theta \le C_1 \operatorname{dist}\Big(0, \partial\rho(\boldsymbol{u})\Big), \ \forall\, \boldsymbol{u} \in B(\overline{\boldsymbol{u}}, \delta).$$

The above KL property (also known as KL inequality) states the regularity of $h(u)$ around its critical point u and the KL inequality trivially holds at a noncritical point. There is a very large set of functions satisfying the KL inequality, including any semi-algebraic functions [2]. Clearly, the objective function f is semi-algebraic as both h and $\delta_{\mathbb{U}}$ and $\delta_{\mathbb{V}}$ are semi-algebraic.

Lemma 1 (Uniform KL property). *There exist $\delta_0 > 0$, $\theta_{KL} \in [0, 1)$, $C_{KL} > 0$ such that for all W s.t.* $\operatorname{dist}((W), \mathbb{C}(W_0)) \le \delta_0$

$$\left|f(W) - \overline{f}\right|^{\theta_{KL}} \le C_{KL} \operatorname{dist}\Big(0, \partial f(W)\Big), \tag{37}$$

with $\overline{f}$ denoting the limiting function value defined in (P2) of Theorem 1.

Proof. First, we recognize the union $\bigcup_i B(W_i^\star, \delta_i)$ forms an open cover of $\mathbb{C}(W_0)$ with $W_i^\star$ representing all points in $\mathbb{C}(W_0)$ and δ_i to be chosen so that the the following KL property of f at $W_i^\star \in \mathbb{C}(W_0)$ holds:

$$\left|f(W) - \overline{f}\right|^{\theta_i} \le C_i \operatorname{dist}\Big(0, \partial f(W)\Big) \ \forall\, (W) \in B(W_i^\star, \delta_i),$$

where we have used all $f(W_i^\star) = \overline{f}$ by assertion (P3) of Theorem 1. Then, due to the compactness of the set $\mathbb{C}(W_0)$, it has a finite subcover $\bigcup_{i=1}^{p} B(W_{k_i}^\star, \delta_{k_i})$ for some positive integer p. Now combining all, for all $W \in \bigcup_{i=1}^{p} B(W_{k_i}^\star, \delta_{k_i})$, we have

$$\left|f(W) - \overline{f}\right|^{\theta_{KL}} \leq C_{KL}\, \mathrm{dist}\left(0, \partial f(W)\right), \tag{38}$$

where $\theta_{KL} = \max_{i=1}^{p}\{\theta_{k_i}\}$ and $C_{KL} = \max_{i=1}^{p}\{C_{k_i}\}$. Finally, since $\bigcup_{i=1}^{p} B(W_{k_i}^\star, \delta_{k_i})$ is an open cover of $\mathbb{C}(W_0)$, there exists a sufficiently small number δ_0 so that

$$\left\{(W) : \mathrm{dist}\left(W, \mathbb{C}(W_0)\right) \leq \delta_0\right\} \subset \bigcup_{i=1}^{p} B(W_i^\star, \delta_{k_i}).$$

Therefore, the KL equation holds whenever $\mathrm{dist}(W, \mathbb{C}(W_0)) \leq \delta_0$. □

We now turn to prove Theorem 2. Note that $h(F, G)$ is a KL function since it is an analytical function and $\delta_{\mathbb{F}}$ and $\delta_{\mathbb{G}}$ are also KL functions as they are the indicator functions on the sets $\mathbb{F}$ and $\mathbb{G}$, respectively. Thus, f also satisfies the above KL property. According to Theorem 1 (P3), $W^\star$ is a critical point of f. It then follows from [1, Lemma 1] and the KL property that there exists a sufficiently large k_0 satisfying

$$\left[f(W_k) - f(W^\star)\right]^{\theta} \leq C_2\, \mathrm{dist}\left(0, \partial f(W_k)\right), \tag{39}$$

for all $k \geq k_0$. Now, we construct a concave function $x^{1-\theta}$ for some $\theta \in [0, 1)$ with domain $x > 0$. By the concavity of the function, we have

$$x_2^{1-\theta} - x_1^{1-\theta} \geq (1-\theta)x_2^{-\theta}(x_2 - x_1), \forall x_1 > 0, x_2 > 0.$$

Replacing x_1 with $f(W_{k+1}) - f(W^\star)$ and x_2 with $f(W_k) - f(W^\star)$, we have

$$\begin{aligned}
&\left[f(W_k) - f(W^\star)\right]^{1-\theta} - \left[f(W_{k+1}) - f(W^\star)\right]^{1-\theta} \\
&\geq (1-\theta)\frac{f(W_k) - f(W_{k+1})}{\left[f(W_k) - f(W^\star)\right]^{\theta}} \\
&\geq \frac{\lambda(1-\theta)}{2C_2}\frac{\|W_k - W_{k+1}\|_F^2}{\mathrm{dist}\left(0, \partial f(W_k)\right)} \\
&\geq \frac{\lambda(1-\theta)}{2C_2C_3}\frac{\|W_k - W_{k+1}\|_F^2}{\|W_k - W_{k-1}\|_F} \\
&= \kappa\left(\frac{\|W_k - W_{k+1}\|_F^2}{\|W_k - W_{k-1}\|_F} + \|W_k - W_{k-1}\|_F\right) - \kappa\|W_k - W_{k-1}\|_F \\
&\geq \kappa\Big(2\|W_k - W_{k+1}\|_F - \|W_k - W_{k-1}\|_F\Big),
\end{aligned}$$

where $C_3 := 2L_c + \frac{1}{\mu} + \frac{1}{\lambda}, \kappa := \frac{\lambda(1-\theta)}{2C_2C_3}$. It then follows that

$$\begin{aligned}
&2\|W_k - W_{k+1}\|_F - \|W_k - W_{k-1}\|_F \\
&\leq \beta\left([f(W_k) - f(W^\star)]^{1-\theta} - [f(W_{k+1}) - f(W^\star)]^{1-\theta}\right),
\end{aligned} \tag{40}$$

with $\beta := (\frac{\lambda(1-\theta)}{2C_2C_3})^{-1}$.

Summing the above inequalities up from some $k_1 > k_0$ to infinity yields

$$\sum_{k=k_1}^{\infty} \|W_k - W_{k+1}\|_F \leq \|W_{k_1} - W_{k_1-1}\|_F + \beta\Big[f(W_{k_1}) - f(W^\star)\Big]^{1-\theta}, \tag{41}$$

which implies that

$$\sum_{k=k_1}^{\infty} \|W_k - W_{k+1}\|_F < \infty.$$

Therefore, the sequence $\{W_k\}$ is Cauchy, and hence convergent. Hence, the limit point set $C(W_0)$ is singleton $W^\star$.

Theorem 3 (Convergence Rate). *The convergence rate is at least sublinear.*

Proof of Theorem 3. To show the convergence rate, we first notice that $\{W_k\}$ converges to some point $W^\star$, i.e., $\lim_{k\to\infty} W^k = W^\star$. Then, by making use of the triangle inequality and Equation (41), we have

$$\begin{aligned} \|W_{k_1} - W^\star\|_F = \left\| \sum_{k=k_1}^{\infty} W_k - W_{k+1} \right\|_F &\leq \sum_{k=k_1}^{\infty} \|W_k - W_{k+1}\|_F \\ &\leq \|W_{k_1} - W_{k_1-1}\|_F + \beta\Big[f(W_{k_1}) - f(W^\star)\Big]^{1-\theta}, \end{aligned} \tag{42}$$

which indicates the convergence rate of $W_{k_1} \to W^\star$ is no slower than the speed that $\|W_{k_1} - W_{k_1-1}\|_F + \beta[f(W_{k_1}) - f(W^\star)]^{1-\theta}$ converges to 0.

Moreover, according to Equation (39), we have

$$\begin{aligned} \beta\Big[f(W_{k_1}) - f(W^\star)\Big]^{\theta} &\leq \beta C_2 \operatorname{dist}\Big(0, \partial f(W_{k_1})\Big) \\ &\leq \underbrace{\beta C_2 \left(2L_c + \frac{1}{\mu} + \frac{1}{\lambda}\right)}_{:=\alpha} \|W_{k_1} - W_{k_1-1}\|_F. \end{aligned} \tag{43}$$

Plugging (43) back to (42), we then have

$$\sum_{k=k_1}^{\infty} \|W_k - W_{k+1}\|_F \leq \|W_{k_1} - W_{k_1-1}\|_F + \alpha \|W_{k_1} - W_{k_1-1}\|_F^{\frac{1-\theta}{\theta}}. \tag{44}$$

Now, we divide the convergence rate analysis into different cases based on the value of KL exponent θ:

— *Case I*: $\theta \in [0, \frac{1}{2}]$, which indicates $\frac{1-\theta}{\theta} \geq 1$. Now define $R_k = \sum_{i=k}^{\infty} \|W_i - W_{i+1}\|_F$, and according to the above, we have

$$R_{k_1} \leq R_{k_1-1} - R_{k_1} + \alpha \left[R_{k_1-1} - R_{k_1}\right]^{\frac{1-\theta}{\theta}}. \tag{45}$$

Since $R_{k_1-1} - R_{k_1} \to 0$, when $k_1 \to \infty$, thus there exists a positive integer $\bar{k}$ such that $R_{k_1-1} - R_{k_1} < 1$, $\forall\, k_1 \geq \bar{k}$. Thus,

$$R_{k_1} \leq (1+\alpha)\,(R_{k_1-1} - R_{k_1}), \quad \forall\, k_1 \geq \bar{k},$$

which implies that

$$R_{k_1} \leq \rho \cdot R_{k_1-1}, \quad \forall\, k_1 \geq \bar{k}, \tag{46}$$

where $\rho = \frac{1+\alpha}{2+\alpha} \in (0,1)$. This together with (42) gives the linear convergence rate

$$\|W_k - W^\star\|_F \leq O(\rho^{k-\bar{k}}), \ \forall\, k \geq \bar{k}. \tag{47}$$

— *Case II*: $\theta \in (1/2, 1)$, which indicates $\frac{1-\theta}{\theta} \leq 1$. According to the former results, we have

$$R_{k_1} \leq (1+\alpha)\left[R_{k_1-1} - R_{k_1}\right]^{\frac{1-\theta}{\theta}}, \quad \forall\ k_1 \geq \bar{k},$$

which is a similar situation to that in [1], where we have

$$R_{k_1}^{\frac{1-2\theta}{1-\theta}} - R_{k_1-1}^{\frac{1-2\theta}{1-\theta}} \geq \zeta,\ \forall\, k \geq \overline{k},$$

for some $\zeta > 0$. By repeating and summing up the above inequalities, we can obtain

$$R_{k_1} \leq \left[R_{k_1-1}^{\frac{1-2\theta}{1-\theta}} + \zeta(k_1 - \overline{k})\right]^{-\frac{1-\theta}{2\theta-1}} = O\left((k_1 - \overline{k})^{-\frac{1-\theta}{2\theta-1}}\right),$$

which indicates the sublinear convergence rate holds.

We end this part by pointing out that the convergence rate is closely related to the KL exponent θ and at least a sublinear convergence rate can be guaranteed. □

4 A NORM-BOUNDED VARIATION

Based on the previous procedure, we can also consider the norm constraint on the factor matrices, instead of just the element-wise bounded constraints. For example, we may be interested in bounding the magnitude of F and G as a whole:

$$\begin{aligned} \mathbb{F} = \mathbb{F}_2 &:= \{F, F \geq 0, \|F\|_F \leq c_1\}, \\ \mathbb{G} = \mathbb{G}_2 &:= \{G, G \geq 0, \|G\|_F \leq c_2\}. \end{aligned} \tag{48}$$

Algorithm 1 still holds for the updating with the following projection:

$$\mathcal{P}_{\mathbb{F}}(F) = \begin{cases} \max(F, 0), & \|\max(F, 0)\|_F \leq c_1, \\ c_1 \frac{\max(F, 0)}{\|\max(F, 0)\|_F}, & \|\max(F, 0)\|_F > c_1. \end{cases} \tag{49}$$

And the proof is almost the same as the element constraint except in Equation (16), where we have $L_c = 2(c_1^2 + c_2^2 + c_1 c_2 + \|X\|_F)$ when $\mathbb{F} = \mathbb{F}_2, \mathbb{G} = \mathbb{G}_2$.

For the analysis of sequence convergence of MUA, under mild conditions, we suppose (1) the sequences F and G are norm bounded; thus, Proposition 1 holds; and (2) during every update, $\lambda = \frac{F_{ij}}{(FGG^T)_{ij}} \in (\underline{\lambda}, \min(\overline{\lambda}, \frac{1}{L_k}))$ (similar for μ), Equation (24) and Equation (25) hold simultaneously. By summing over all elements in F and G, Equation (18) and Equation (19) hold, and so are the theorems; thus, the sequence is convergent.

5 EXPERIMENTS

In this section, we empirically study the performance of our proposed *factor-bounded NMF* method using several benchmark datasets and compare our method against a collection of state-of-the-art NMF methods. Our experiments show that the bounding constraints on the factor matrices of F and G can provide a favorable representation for the original data X in a learned space and such a representation can be further utilized to improve the clustering performance. Our experiment sets the dimensionality of the learned feature space as the number of classes of the original dataset. For example, in the AT&T Faces dataset, we set r equal to 40 for each selected algorithm. Once we get the learned representations, in which n columns are involved in G, we use two approaches to determine the clusters for this reduced representation. One approach is the K-Means algorithm and another approach is G-indicator, which explores the clustering results directly on the partition matrix G [10]. These experimental approaches are inspired by the work done by Liu et al. [24, 35]. All hyper-parameters associated with the methods reported in Tables 1–6 are found via a reasonably sized grid search performed for each method and dataset pair.

Table 1. Adjusted Rand Index and Standard Deviations (on 10 Runs) for an Array of Matrix Factorization Methods Tested on Various Real-World Datasets Compared to the Proposed Factor-Bounded NMF

	K-means	NMF	LSNMF	nsNMF	PMF	SNMF	ANMF	G-inidicator	Ours
Heart	0.089	0.101 ± 0.047	0.131 ± 0.004	**0.236 ± 0.124**	0.213 ± 0.106	0.188 ± 0.071	0.217±0.026	0.129± 0.011	0.201 ± 0.059
Ionosphere	0.015	0.060 ± 0.007	0.066 ± 0.008	0.067 ± 0.015	0.071 ± 0.008	0.077 ± 0.025	0.030 ± 0.015	0.027 ± 0.021	**0.479 ± 0.014**
Wine	0.371	0.743 ± 0.035	0.750 ± 0.018	0.738 ± 0.033	0.751 ± 0.013	0.755 ± 0.012	0.826 ± 0.009	0.749 ± 0.016	**0.857 ± 0.024**
Iris	0.433	0.510 ± 0.012	**0.781 ± 0.151**	0.589 ± 0.145	0.504 ± 0.003	0.563 ± 0.119	0.600 ±0.011	0.598 ± 0.021	0.618 ± 0.078
Cancer	0.492	0.562 ± 0.005	0.567 ± 0.000	0.610 ± 0.015	0.595 ± 0.012	0.569 ± 0.026	0.718 ± 0.029	0.677 ± 0.025	**0.746 ± 0.025**
AT&T Faces	0.387	0.427 ± 0.020	0.497 ± 0.030	0.433 ± 0.016	0.432 ± 0.031	0.476 ± 0.027	0.509 ± 0.011	0.497 ± 0.020	**0.563 ± 0.037**
Mnist	0.598	0.672 ± 0.004	0.728 ± 0.010	0.695 ± 0.011	0.601 ± 0.007	0.675 ± 0.020	0.734 ± 0.005	0.603 ± 0.015	**0.772 ± 0.013**
USPS	0.399	0.427 ± 0.020	0.497 ± 0.030	0.433 ± 0.016	0.432 ± 0.031	0.476 ± 0.027	**0.527 ± 0.016**	0.472 ± 0.036	0.513 ± 0.037

The cluster membership of each data point is determined by generating a reduced representation ($G \in \mathbb{R}^{r \times n}$) that is fed into a standard K-Means algorithm. We set r equal the number of classes contained in each dataset.

We compare our method to the following six matrix factorization methods implemented in Python [41]. First, the standard multiplicative-update NMF [20] method, which is widely used to determine clusters in text analysis, image processing, and bioinformatics applications. Second, an implementation of **Least-Squares NMF** (**LSNMF**) using a projected gradient method [22] that has shown faster convergence than the multiplicative update method. Third, the **Nonsmooth Nonnegative Matrix Factorization** (**nsNMF**) proposed by Pascual-Montano et al. [26], which is able to identify localized features from the input data through an objective that promotes sparsity. Fourth, a **Probabilistic NMF** (**PMF**) method motivated by the analysis of Laurberg et al. [18], which argues for a probabilistic interpretation of matrix factorization. Fifth, we compare our method to a **Sparse NMF** (**SNMF**) that uses an alternating nonnegative LS approach [16]. This approach is designed to impose a controllable sparsity, via hyper-parameter tuning, on the learned F and G matrices. Finally, the **Adversarial NMF** (**ANMF**) approach, which considers potential test adversaries that are beyond the pre-defined constraints, instead of only focusing on the regular data points [25].

5.1 Clustering Performance

In Tables 1–3, we, respectively, present the average adjusted rand scores, **normalized mutual information** (**NMI**), and **accuracy classification scores** (**ACCs**), with their standard deviations for our method and aforementioned matrix factorization algorithms. We test the performance of our method on a collection of widely used datasets downloaded from the UCI machine learning dataset repository[1] and two large datasets. The datasets are chosen to illustrate the versatility that our factor-bounded NMF approach provides a clustering algorithm in a variety of problem domains. In each experiment, the data are normalized and the bounds on F and G are set accordingly from 0 to 255 for imaging datasets and 0 to 1 for others.

From Table 1, we can see that our proposed method is effective at creating a reduced representation that can be used to accurately cluster data for the majority of the chosen datasets. Our method outperforms other matrix factorization methods with clear margins, in terms of clustering performance, on the *Ionosphere*, *Wine*, *Cancer*, *AT&T Faces*, and *Mnist* datasets. In particular, the clustering adjusted rand scores of our learned representation of the *Ionosphere* dataset is significantly more effective than other compared NMF algorithms. In Tables 2 and 3, we find that our method presents better NMI and ACC for the most selected datasets. Even though the ANMF outperforms our method in some cases, applying ANMF on large datasets is limited to the bottleneck of slow computational speed because of intensive matrix multiplications involved in each iteration step of

[1] http://archive.ics.uci.edu/ml/index.php.

Table 2. NMI and Standard Deviations (on 10 Runs) for an Array of Matrix Factorization Methods Tested on Various Real-World Datasets Compared to the Proposed Factor-Bounded NMF

	K-means	NMF	LSNMF	nsNMF	PMF	SNMF	ANMF	G-inidicator	Ours
Heart	0.021	0.019 ± 0.021	0.052 ± 0.039	0.146 ± 0.005	0.131 ± 0.074	0.119 ± 0.062	0.116 ± 0.009	0.098 ± 0.010	**0.161 ± 0.036**
Ionosphere	0.037	0.066 ± 0.010	0.070 ± 0.009	0.071 ± 0.005	0.043 ± 0.017	0.040 ± 0.012	0.074 ± 0.004	0.069 ± 0.021	**0.082 ± 0.025**
Wine	0.410	0.518 ± 0.020	**0.672 ± 0.038**	0.607 ± 0.016	0.587 ± 0.030	0.588 ± 0.025	0.654 ± 0.026	0.623 ± 0.070	0.637 ± 0.032
Iris	0.672	0.682 ± 0.040	0.899 ± 0.009	0.871 ± 0.042	0.682 ± 0.010	0.888 ± 0.042	0.906 ± 0.004	0.796 ± 0.016	**0.913 ± 0.028**
Cancer	0.421	0.452 ± 0.009	0.501 ± 0.032	0.572 ± 0.011	0.549 ± 0.080	0.477 ± 0.036	0.654 ± 0.005	0.503 ± 0.009	**0.698± 0.012**
AT&T Faces	0.331	0.537 ± 0.003	0.570 ± 0.010	0.533 ± 0.019	0.609 ± 0.014	0.611 ± 0.014	0.591 ± 0.017	0.488 ± 0.009	**0.629 ± 0.030**
Mnist	0.358	0.348 ± 0.027	0.401 ± 0.009	0.529 ± 0.037	0.601 ± 0.004	0.653 ± 0.025	0.742 ± 0.045	0.701 ± 0.002	**0.793 ± 0.021**
USPS	0.466	0.657 ± 0.020	0.597 ± 0.023	0.633 ± 0.006	0.637 ± 0.001	0.701 ± 0.017	**0.721 ± 0.017**	0.588 ± 00.021	0.709 ± 0.009

The cluster membership of each data point is determined by generating a reduced representation ($G \in \mathbb{R}^{r \times n}$) that is fed into a standard K-Means algorithm. We set r equal the number of classes contained in each dataset.

Table 3. ACCs and Standard Deviations (on 10 Runs) for an Array of Matrix Factorization Methods Tested on Various Real-World Datasets Compared to the Proposed Factor-Bounded NMF

	K-means	NMF	LSNMF	nsNMF	PMF	SNMF	ANMF	G-inidicator	Ours
Heart	0.020	0.051 ± 0.039	0.041 ± 0.005	0.144 ± 0.074	0.119 ± 0.062	0.025 ± 0.018	**0.210 ± 0.007**	0.101 ± 0.011	0.197 ± 0.036
Ionosphere	0.031	0.091 ± 0.009	0.049 ± 0.001	0.081 ± 0.011	0.039 ± 0.012	0.083 ± 0.006	0.083 ± 0.017	0.076 ± 0.027	**0.086 ± 0.025**
Wine	0.328	0.631 ± 0.047	0.630 ± 0.015	0.647 ± 0.03	0.656 ± 0.036	0.656 ± 0.046	0.722 ± 0.001	0.698 ± 0.026	**0.731 ± 0.044**
Iris	0.664	0.902 ± 0.045	0.864 ± 0.048	0.674 ± 0.019	0.880 ± 0.055	0.908 ± 0.019	0.908 ± 0.028	0.866 ± 0.010	**0.910 ± 0.009**
Cancer	0.450	0.520 ± 0.063	0.510 ± 0.015	0.477 ± 0.076	0.490 ± 0.084	0.505 ± 0.045	**0.712 ± 0.020**	0.509 ± 0.007	0.697 ± 0.012
AT&T Faces	0.308	0.521 ± 0.013	0.582 ± 0.034	0.527 ± 0.022	0.572 ± 0.026	0.545 ± 0.026	0.542 ± 0.0168	0.422 ± 0.012	**0.606 ± 0.018**
Mnist	0.326	0.382 ± 0.063	0.322 ± 0.015	0.372 ± 0.076	0.427 ± 0.084	0.453 ± 0.045	0.812 ± 0.0328	0.601 ± 0.023	**0.820 ± 0.012**
USPS	0.427	0.749 ± 0.030	0.503 ± 0.011	0.621 ± 0.010	0.602 ± 0.023	0.701 ± 0.003	0.734 ± 0.005	0.501 ± 0.023	**0.751 ± 0.037**

The cluster membership of each data point is determined by generating a reduced representation ($G \in \mathbb{R}^{r \times n}$) that is fed into a standard K-Means algorithm. We set r equal the number of classes contained in each dataset.

the solution algorithms. The results from Tables 1–3 show that our method can be used as an effective low-dimensional embedding and afford accurate clustering results for a variety of datasets.

Robustness to Noise. In order to further validate the usefulness of our proposed method, we perform the same experiment as reported in Table 1 but add random (uniform) noise to each element contained in X. The Gaussian noise added to each individual element is randomly distributed with the variance of 0.01. Once the random noise has been added, the experiment follows the same flow as described above; the resulting data are normalized and they are fed into a corresponding NMF algorithm to calculate a reduced representation G, which is then used as an input into a standard K-Means algorithm. The results of this experiment are meant to illustrate that our method can produce reasonable results even when random noise is incorporated into the data.

In Tables 4–6, we report the results of our method compared against various other NMF algorithms when the original data are artificially corrupted. We find that our factor-bounded NMF method appears to generate an effective reduced representation of X that can be used to cluster the data even though a significant amount of random noise is added into the data, which indicates the potential applications of our method, especially for dealing with noisy data. The authors remark that this robustness to noisy data could be enhanced if we considered an implementation of the nonsquared Frobenius-norm instead of the squared Frobenius-norm defined in Equation (7). Nonetheless, our proposed algorithm learns a representation that can more accurately cluster noisy data points when compared to other state-of-the-art NMF approaches.

5.2 Empirical Convergence

In Section 3, we analyzed the convergence of our projected gradient descent algorithm. Here, we provide some empirical evidence to illustrate the practical convergence of our approach. In the

Table 4. The Calculated Adjusted Rand Index and Standard Deviations of a Collection of NMF Algorithms Performed on Real-World Datasets that Are Corrupted with *Random Noise*

	K-means	NMF	LSNMF	nsNMF	PMF	SNMF	ANMF	G-inidicator	Ours
Heart	0.072	0.088 ± 0.039	0.117 ± 0.005	0.092 ± 0.074	0.109 ± 0.062	0.118 ± 0.018	**0.180 ± 0.009**	0.129 ± 0.023	0.173 ± 0.036
Ionosphere	0.011	0.017 ± 0.009	0.015 ± 0.001	0.015 ± 0.011	0.021 ± 0.012	0.021 ± 0.006	0.106 ± 0.003	0.201 ± 0.002	**0.216 ± 0.025**
Wine	0.169	0.219 ± 0.047	0.232 ± 0.015	0.184 ± 0.03	0.189 ± 0.036	0.204 ± 0.046	0.198 ± 0.025	0.211 ± 0.017	**0.237 ± 0.044**
Iris	0.209	0.213 ± 0.045	0.317 ± 0.048	0.222 ± 0.019	0.211 ± 0.055	0.340 ± 0.019	0.306 ± 0.014	0.299 ± 0.007	**0.349 ± 0.009**
Cancer	0.069	0.072 ± 0.063	0.183 ± 0.015	0.161 ± 0.076	0.115 ± 0.084	0.242 ± 0.045	0.374 ± 0.022	0.263 ± 0.011	**0.487 ± 0.012**
AT&T Faces	0.206	0.245 ± 0.013	0.449 ± 0.034	0.238 ± 0.022	0.251 ± 0.026	0.434 ± 0.026	0.326 ± 0.009	0.411 ± 0.024	**0.497 ± 0.018**
Mnist	0.218	0.322 ± 0.002	0.315 ± 0.004	0.287 ± 0.007	0.347 ± 0.010	0.358 ± 0.009	0.217 ± 0.003	0.208 ± 0.011	**0.381 ± 0.014**
USPS	0.102	0.117 ± 0.011	0.208 ± 0.014	0.188 ± 0.009	0.241 ± 0.015	0.187 ± 0.022	**0.281 ± 0.020**	0.200 ± 0.017	0.275 ± 0.021

Following [24], the cluster membership of each noised data point is determined by a K-Means algorithm applied to the reduced representation learned by each NMF algorithm. Identical to the experiment reported in Table 1, we set r equal the number of classes contained in each dataset.

Table 5. NMI and Standard Deviations of a Collection of NMF Algorithms Performed on Real-World Datasets that Are Corrupted with *Random Noise*

	K-means	NMF	LSNMF	nsNMF	PMF	SNMF	ANMF	G-inidicator	Ours
Heart	0.013	0.011 ± 0.007	0.009 ± 0.005	0.015 ± 0.074	0.077 ± 0.062	0.098 ± 0.018	0.107 ± 0.009	0.019 ± 0.023	**0.111 ± 0.036**
Ionosphere	0.015	0.023 ± 0.009	0.034 ± 0.001	0.027 ± 0.011	0.018 ± 0.012	0.018 ± 0.006	**0.030 ± 0.003**	0.027 ± 0.002	0.027 ± 0.025
Wine	0.122	0.306 ± 0.047	0.222 ± 0.015	0.269 ± 0.03	0.249 ± 0.036	0.176 ± 0.046	0.309 ± 0.025	0.301 ± 0.017	**0.311 ± 0.044**
Iris	0.301	0.247 ± 0.045	0.309 ± 0.048	0.231 ± 0.019	0.195 ± 0.055	0.300 ± 0.019	0.328 ± 0.014	0.270 ± 0.007	**0.351 ± 0.009**
Cancer	0.106	0.119 ± 0.063	0.277 ± 0.015	0.241 ± 0.076	0.305 ± 0.084	0.118 ± 0.045	0.307 ± 0.022	0.202 ± 0.011	**0.311 ± 0.012**
AT&T Faces	0.247	0.288 ± 0.013	0.321 ± 0.034	0.242 ± 0.022	0.268 ± 0.026	0.321 ± 0.026	0.319 ± 0.009	0.207 ± 0.024	**0.325 ± 0.018**
Mnist	0.176	0.211 ± 0.010	0.275 ± 0.009	0.301 ± 0.012	0.314 ± 0.004	0.286 ± 0.014	0.299 ± 0.003	0.189 ± 0.008	**0.374 ± 0.006**
USPS	0.217	0.229 ± 0.018	0.306 ± 0.017	0.253 ± 0.010	0.286 ± 0.021	0.294 ± 0.013	**0.317 ± 0.006**	0.262 ± 0.021	0.309 ± 0.027

Following [24], the cluster membership of each noised data point is determined by a K-Means algorithm applied to the reduced representation learned by each NMF algorithm. Identical to the experiment reported in Table 2, we set r equal the number of classes contained in each dataset.

Table 6. ACCs and Standard Deviations of a Collection of NMF Algorithms Performed on Real-World Datasets that Are Corrupted with *Random Noise*

	K-means	NMF	LSNMF	nsNMF	PMF	SNMF	ANMF	G-inidicator	Ours
Heart	0.009	0.011 ± 0.007	0.018 ± 0.003	0.027 ± 0.011	0.047 ± 0.005	0.018 ± 0.009	0.076 ± 0.013	0.030 ± 0.008	**0.088 ± 0.011**
Ionosphere	0.011	0.027 ± 0.009	0.015 ± 0.005	0.027 ± 0.006	0.014 ± 0.003	0.033 ± 0.010	0.029 ± 0.010	0.076 ± 0.011	**0.034 ± 0.021**
Wine	0.118	0.201 ± 0.043	0.228 ± 0.011	0.189 ± 0.030	0.293 ± 0.027	0.285 ± 0.034	0.244 ± 0.09	0.268 ± 0.021	**0.300 ± 0.035**
Iris	0.289	0.256 ± 0.039	0.301 ± 0.050	0.251 ± 0.014	0.205 ± 0.049	0.278 ± 0.023	0.302 ± 0.018	0.222 ± 0.008	**0.341 ± 0.010**
Cancer	0.113	0.201 ± 0.058	0.275 ± 0.014	0.247 ± 0.069	0.296 ± 0.081	0.120 ± 0.034	**0.311 ± 0.034**	0.208 ± 0.010	0.308 ± 0.010
AT&T Faces	0.209	0.271 ± 0.011	0.309 ± 0.028	0.231 ± 0.017	0.254 ± 0.031	0.287 ± 0.026	0.296 ± 0.018	0.204 ± 0.009	**0.314 ± 0.012**
Mnist	0.188	0.201 ± 0.014	0.269 ± 0.011	0.285 ± 0.008	0.311 ± 0.004	0.295 ± 0.025	0.296 ± 0.005	0.197 ± 0.010	**0.368 ± 0.007**
USPS	0.214	0.218 ± 0.032	0.304 ± 0.014	0.248 ± 0.012	0.277 ± 0.018	0.301 ± 0.006	0.290 ± 0.002	0.209 ± 0.022	**0.320 ± 0.038**

Following [24], the cluster membership of each noised data point is determined by a K-Means algorithm applied to the reduced representation learned by each NMF algorithm. Identical to the experiment reported in Table 3, we set r equal the number of classes contained in each dataset.

following figures, we are interested in showing two types of convergence. First, we would like to verify that our algorithm obtains an overall convergence of our objective and, second, that F and G also converge. This experiment serves as a validation of the rigorous mathematical convergence analysis of our algorithm.

Figure 2 shows how the objective in Equation (7) and MUA changes through an update on the *Wine* dataset. We can clearly see that the objective function using our method decreases at a sub-linear rate, which presents a faster converging rate relative to the MUA method. This finding successfully supports our previous mathematical discussions. Furthermore, in order to validate

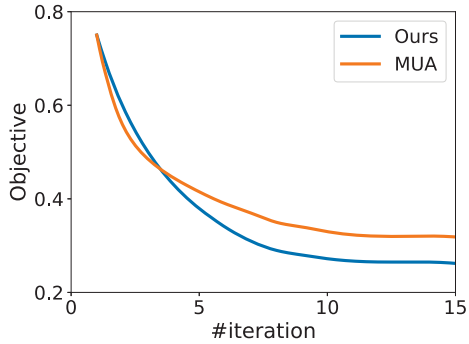

Fig. 2. The converging rate between the factor-bounded NMF objective (Equation (7)) and MUA.

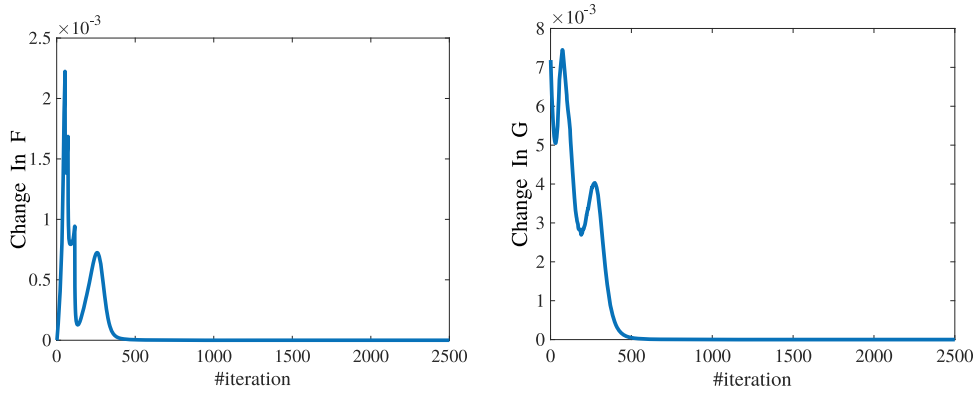

Fig. 3. **Left**: $\|F_{k+1} - F_k\|_F$ plotted with an update of our method. **Right**: $\|G_{k+1} - G_k\|_F$ plotted with an update. Both plots converge to zero after several iterations; this empirically validates the convergence of the matrix factors discussed in Section 3.

that our proposed method gets factor matrices F and G converged, we plot the gap between every two iterations in Figure 3. We see that after around 500 iterations, both factor matrices are stable and have converged; this empirical evidence also agrees with our proof.

5.3 Case Study: Learned Features

In the seminal paper introducing NMF by Lee and Seung [19], the authors stress that an interpretable benefit of NMF is that it can learn a "parts-based representation" of the original data instead of the "holistic" representations, which are learned via algorithms such as **principal component analysis** (**PCA**) and other similar embeddings. Provided that the bounds on F and G are also positive, we would assume that our algorithm will also learn a parts-based representation. To verify this assumption, and see how our method's learned features compare to other NMF algorithms, we plot a few components of each matrix factorization algorithm derived from the rows of F trained on the *AT&T Faces* dataset.

In Figure 4, we plot five randomly chosen rows of F for the same algorithms tested in Tables 1 and 4. We clearly see in Figure 4 that our proposed algorithm learns a parts-based representation similar to other NMF implementations. This "parts-based representation," provided in conjunction with the factor-bounded constraint, affords a level of guaranteed interpretability that, to the authors knowledge, has not previously been shown.

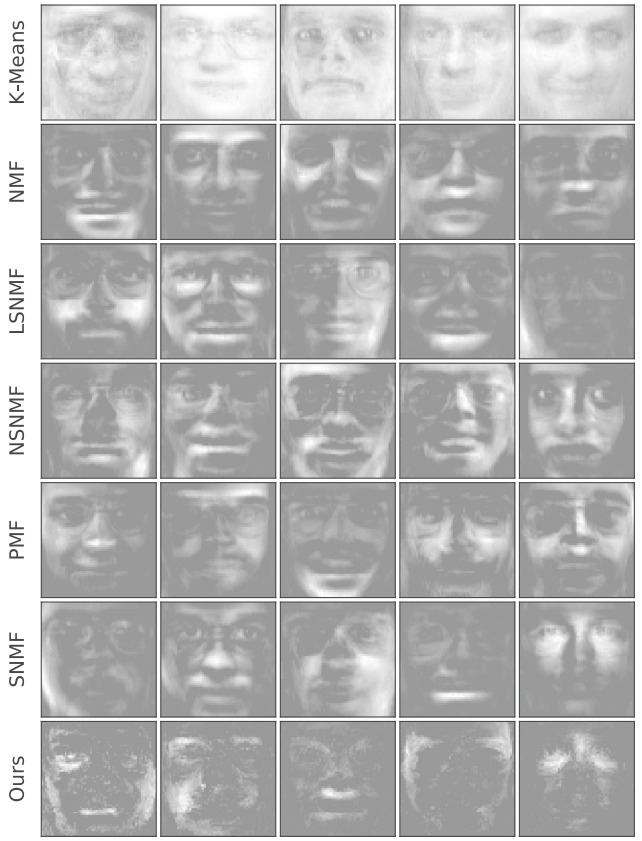

Fig. 4. Learned feature comparison between our method and other state-of-the-art methods on the *AT&T Faces* dataset. Note that we only show 5 randomly chosen (out of 40) centroids selected from the rows of F for each method. Observe that the factor-bounded NMF algorithm also learns a "parts-based representation" similar to other compared algorithms.

6 CONCLUSION

In this article, we explored factor-bounded NMF problems where the feature matrix and the membership matrix are assumed to lie within certain regions. In particular, we incorporated the element-wise constraint on the factor matrices to make the NMF more suitable for clustering applications. Moreover, we have provided a rigorous convergence analysis for the alternating projected gradient descent when applied to solve the corresponding factor-bounded NMF. We have shown the sequence generated by the projected gradient descent method is convergent and that it converges to a critical point.

REFERENCES

[1] Hedy Attouch and Jérôme Bolte. 2009. On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Mathematical Programming* 116, 1-2 (2009), 5–16.

[2] Hedy Attouch, Jérôme Bolte, and Benar Fux Svaiter. 2013. Convergence of descent methods for semi-algebraic and tame problems: Proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods. *Mathematical Programming* 137, 1-2 (2013), 91–129.

[3] Jérôme Bolte, Aris Daniilidis, and Adrian Lewis. 2007. The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM Journal on Optimization* 17, 4 (2007), 1205–1223.

[4] Deng Cai, Xiaofei He, Jiawei Han, and Thomas S. Huang. 2011. Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 8 (2011), 1548–1560.
[5] Paul H. Calamai and Jorge J. Moré. 1987. Projected gradient methods for linearly constrained problems. *Mathematical Programming* 39, 1 (1987), 93–116.
[6] Mulin Chen, Qi Wang, and Xuelong Li. 2018. Adaptive projected matrix factorization method for data clustering. *Neurocomputing* 306 (2018), 182–188. DOI: https://doi.org/10.1016/j.neucom.2018.04.031
[7] Moody Chu, Fasma Diele, Robert Plemmons, and Stefania Ragni. 2004. Optimality, computation, and interpretation of nonnegative matrix factorizations. *SIAM Journal on Matrix Analysis* (2004). DOI: 10.1.1.61.5758&rep=rep1&type=pdf
[8] Chris Ding, Xiaofeng He, and Horst D. Simon. 2005. On the equivalence of nonnegative matrix factorization and spectral clustering. In *Proceedings of the 2005 SIAM International Conference on Data Mining*. SIAM, 606–610.
[9] Chris Ding, Tao Li, and Michael I. Jordan. 2010. Convex and semi-nonnegative matrix factorizations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32, 1 (2010), 45–55.
[10] Chris Ding, Tao Li, Wei Peng, and Haesun Park. 2006. Orthogonal nonnegative matrix t-factorizations for clustering. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 126–135.
[11] Hongchang Gao, Feiping Nie, Weidong Cai, and Heng Huang. 2015. Robust capped norm nonnegative matrix factorization: Capped norm NMF. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. ACM, 871–880.
[12] Quanquan Gu and Jie Zhou. 2009. Co-clustering on manifolds. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 359–368.
[13] Xiaofei He and Partha Niyogi. 2004. Locality preserving projections. In *Proceedings of the 16th International Conference on Neural Information Processing*. ACM, 153–160.
[14] Patrik O. Hoyer. 2004. Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research* 5 (2004), 1457–1469. https://jmlr.org/papers/volume5/hoyer04a/hoyer04a.pdf.
[15] Kejun Huang, Nicholas D. Sidiropoulos, and Athanasios P. Liavas. 2016. A flexible and efficient algorithmic framework for constrained matrix and tensor factorization. *IEEE Transactions on Signal Processing* 64, 19 (2016), 5052–5065.
[16] Hyunsoo Kim and Haesun Park. 2007. Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics* 23, 12 (2007), 1495–1502.
[17] Deguang Kong, Chris Ding, and Heng Huang. 2011. Robust nonnegative matrix factorization using l21-norm. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*. ACM, 673–682.
[18] Hans Laurberg, Mads Græsbøll Christensen, Mark D. Plumbley, Lars Kai Hansen, and Søren Holdt Jensen. 2008. Theorems on positive data: On the uniqueness of NMF. *Computational Intelligence and Neuroscience* 2008 (2008), 764206. https://doi.org/10.1155/2008/764206
[19] Daniel D. Lee and H. Sebastian Seung. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature* 401, 6755 (1999), 788.
[20] Daniel D. Lee and H. Sebastian Seung. 2001. Algorithms for non-negative matrix factorization. In *Proceedings of the 13th International Conference on Neural Information Processing Systems*. ACM, 556–562.
[21] Xuelong Li, Mulin Chen, and Qi Wang. 2019. Discrimination-aware projected matrix factorization. *IEEE Transactions on Knowledge and Data Engineering* 32, 4 (2019), 809–814.
[22] Chih-Jen Lin. 2007. Projected gradient methods for nonnegative matrix factorization. *Neural Computation* 19, 10 (2007), 2756–2779.
[23] Kai Liu and Hua Wang. 2015. Robust multi-relational clustering via l1-norm symmetric nonnegative matrix factorization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Vol. 2. 397–401.
[24] Weixiang Liu, Kehong Yuan, and Datian Ye. 2008. Reducing microarray data via nonnegative matrix factorization for visualization and clustering analysis. *Journal of Biomedical Informatics* 41, 4 (2008), 602–606.
[25] Lei Luo, Yanfu Zhang, and Heng Huang. 2020. Adversarial nonnegative matrix factorization. In *Proceedings of the 37th International Conference on Machine Learning*. PMLR, 6479–6488.
[26] Alberto Pascual-Montano, Jose Maria Carazo, Kieko Kochi, Dietrich Lehmann, and Roberto D. Pascual-Marqui. 2006. Nonsmooth nonnegative matrix factorization (nsNMF). *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 3 (2006), 403–415.
[27] Anh Huy Phan and Andrzej Cichocki. 2008. Multi-way nonnegative tensor factorization using fast hierarchical alternating least squares algorithm (HALS). In *Proceedings of the 2008 International Symposium on Nonlinear Theory and its Applications*.
[28] Meisam Razaviyayn, Mingyi Hong, and Zhi-Quan Luo. 2013. A unified convergence analysis of block successive minimization methods for nonsmooth optimization. *SIAM Journal on Optimization* 23, 2 (2013), 1126–1153.

[29] Mikkel N. Schmidt and Rasmus K. Olsson. 2006. Single-channel speech separation using sparse non-negative matrix factorization. In *Proceedings of the 9th International Conference on Spoken Language Processing*.

[30] Martin Slawski, Matthias Hein, and Pavlo Lutsik. 2013. Matrix factorization with binary components. *Advances in Neural Information Processing Systems* 26 (2013), 3210–3218. https://papers.nips.cc/paper/2013/file/226d1f15ecd35f784d2a20c3ecf56d7f-Paper.pdf.

[31] Paris Smaragdis and Judith C. Brown. 2003. Non-negative matrix factorization for polyphonic music transcription. In *Proceedings of the 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 177–180.

[32] Norikazu Takahashi and Masato Seki. 2016. Multiplicative update for a class of constrained optimization problems related to NMF and its global convergence. In *Proceedings of the 2016 24th European Signal Processing Conference*. IEEE, 438–442.

[33] Hua Wang, Heng Huang, Chris Ding, and Feiping Nie. 2013. Predicting protein–protein interactions from multimodal biological data sources via nonnegative matrix tri-factorization. *Journal of Computational Biology* 20, 4 (2013), 344–358.

[34] Hua Wang, Feiping Nie, Heng Huang, and Chris Ding. 2011. Nonnegative matrix tri-factorization based high-order co-clustering and its fast implementation. In *Proceedings of the 2011 IEEE 11th International Conference on Data Mining,*. IEEE, 774–783.

[35] Hua Wang, Feiping Nie, Heng Huang, and Fillia Makedon. 2011. Fast nonnegative matrix tri-factorization for large-scale data co-clustering. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*.

[36] Wei Xu, Xin Liu, and Yihong Gong. 2003. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 267–273.

[37] Yangyang Xu and Wotao Yin. 2013. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Sciences* 6, 3 (2013), 1758–1789.

[38] Felipe Yanez and Francis Bach. 2017. Primal-dual algorithms for non-negative matrix factorization with the Kullback–Leibler divergence. In *Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing,*. IEEE, 2257–2261.

[39] Rafal Zdunek. 2008. Data clustering with semi-binary nonnegative matrix factorization. In *Proceedings of the International Conference on Artificial Intelligence and Soft Computing*. Springer, 705–716.

[40] Xiang Zhang, Naiyang Guan, Long Lan, Dacheng Tao, and Zhigang Luo. 2014. Box-constrained projective nonnegative matrix factorization via augmented Lagrangian method. In *Proceedings of the 2014 International Joint Conference on Neural Networks*. IEEE, 1900–1906.

[41] Marinka Žitnik and Blaž Zupan. 2012. NIMFA: A Python library for nonnegative matrix factorization. *Journal of Machine Learning Research* 13, (2012), 849–853. https://www.jmlr.org/papers/volume13/zitnik12a/zitnik12a.pdf.

Received July 2020; revised January 2021; accepted February 2021