2023 International Symposium on Nonlinear Theory and Its Applications
NOLTA2023, September 26-29, 2023, Catania and Online

A Novel HALS-Based Iterative Algorithm for Randomized Nonnegative Matrix Factorization

Takao Masuda†, Kento Yamada‡, Tsuyoshi Migita⋆ and Norikazu Takahashi⋆

† Graduate School of Natural Science and Technology, Okayama University
‡ Graduate School of Environmental, Life, Natural Science and Technology, Okayama University
⋆ Faculty of Environmental, Life, Natural Science and Technology, Okayama University
3–1–1 Tsushima-naka, Kita-ku, Okayama 700-8530 Japan

Email: { pk396rxo@s. | pj3w250a@s. | migita-t@ | takahashi@ } okayama-u.ac.jp

Abstract— As an approach to efficiently perform large-scale Nonnegative Matrix Factorization (NMF), randomized NMF was recently proposed. This approach reduces the dimension of a given nonnegative matrix by multiplying it by a random matrix, and then performs matrix factorization. However, algorithms for standard NMF cannot be directly used in randomized NMF because linear inequality constraints have to be considered instead of nonnegativity ones. In this paper, we reformulate the optimization problem for randomized NMF from a different perspective and propose a novel iterative algorithm which is a combination of the hierarchical alternating least squares algorithm and projection onto the feasible region. We also prove that the proposed algorithm has global convergence property.

1. Introduction

Nonnegative Matrix Factorization (NMF) is the mathematical operation that decomposes a given nonnegative matrix into two low-rank nonnegative matrices [1, 2]. Since NMF can extract latent features in the given nonnegative matrix as a set of nonnegative vectors, it has been applied to image processing, audio signal processing, text mining, recommender systems, network analysis, and so on. NMF is formulated as an optimization problem with nonnegativity constraits [1]. Multiplicative Update Rule (MUR) [1] and Hierarchical Alternating Least Squares (HALS) algorithms [3] are widely used for solving such problems. However, when the given nonnegative matrix is very large, these algorithms require long computation time.

As an approach to efficiently perform large-scale NMF, randomized NMF was recently proposed [5]. In this approach, a given nonnegative matrix is multiplied by a random matrix to obtain a new matrix of lower dimension, and then this new matrix is decomposed into two low-rank matrices. However, algorithms for standard NMF cannot be directly used in randomized NMF because some of the nonnegativity constrains in the original optimization problem are transformed into linear inequality constraints.

ORCID iDs Takao Masuda: 0009-0006-4965-1682, Kento Yamada: 0009-0005-7269-0375, Tsuyoshi Migita: 0009-0006-3242-2090, Norikazu Takahashi: 0000-0001-8222-5593

An algorithm for randomized NMF was first proposed by Erichson *et al.* [5]. However, their algorithm does not exactly solve the optimization problem for randomized NMF. In fact, it does not always return a solution that satisfies the linear inequality constraints. Another algorithm for randomized NMF was proposed by the authors [6]. Although this algorithm has been proved to be globally convergent to the set of stationary points, it is slow because only one variable is updated at a time.

In this paper, we reformulate the optimization problem for randomized NMF from a different perspective, and propose a novel iterative algorithm which is a combination of the HALS algorithm and projection onto the feasible region. We also show through experiments using two real datasets that the proposed algorithm is valid and effective.

2. Randomized NMF

NMF is the operation that decomposes a given nonnegative matrix $\boldsymbol{X} \in \mathbb{R}_+^{M\times N}$ into two nonnegative matrices $\boldsymbol{W} \in \mathbb{R}_+^{M\times K}$ and $\boldsymbol{H} \in \mathbb{R}_+^{N\times K}$ that satisfy $\boldsymbol{X} \approx \boldsymbol{W}\boldsymbol{H}^{\mathrm{T}}$, where $\mathbb{R}_+$ denotes the set of nonnegative real numbers. NMF is formulated as an optimization problem in which the error between $\boldsymbol{X}$ and $\boldsymbol{W}\boldsymbol{H}^{\mathrm{T}}$ is minimized subject to the constraint that all entries of $\boldsymbol{W}$ and $\boldsymbol{H}$ are nonnegative [1]. In the case where the error is measured in the Frobenius norm $\|\cdot\|_{\mathrm{F}}$, the optimization problem for NMF is expressed as follows:

$$\begin{aligned} &\text{minimize} && f(\boldsymbol{W},\boldsymbol{H}) = \frac{1}{2}\left\|\boldsymbol{X}-\boldsymbol{W}\boldsymbol{H}^{\mathrm{T}}\right\|_{\mathrm{F}}^2 \\ &\text{subject to} && \boldsymbol{W}\geq \boldsymbol{0}_{M\times K},\ \boldsymbol{H}\geq \boldsymbol{0}_{N\times K} \end{aligned} \tag{1}$$

where $\boldsymbol{0}_{I\times J}$ denotes the $I\times J$ zero matrix, and the inequality between matrices is applied entry-wise.

Since the objective function $f(\boldsymbol{W},\boldsymbol{H})$ of the problem (1) is not jointly convex, it is difficult in general to find an optimal solution. However, if we consider $\boldsymbol{W}$ or $\boldsymbol{H}$ as constant, the objective function becomes convex. For this reason, many existing algorithms for solving (1) such as the MUR [1] and the HALS algorithm [3] do not update entries of $\boldsymbol{W}$ and $\boldsymbol{H}$ at the same time.

In the case where the matrix $\boldsymbol{X} \in \mathbb{R}_+^{M\times N}$ is very large, the above-mentioned algorithms require long computation

time. In order to overcome this issue, Erichson *et al*. proposed a framework called randomized NMF [5], which is based on approximate matrix decompositions. In the first step of the framework, one generates a random matrix $\boldsymbol{\Omega} \in \mathbb{R}^{N\times K}$ whose entries are drawn i.i.d. from the standard normal distribution or the uniform distribution on $[0, 1]$, and obtains $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\Omega} \in \mathbb{R}^{M\times K}$, where K is an integer smaller than $\min\{M, N\}$ and $\mathbb{R}$ denotes the set of real numbers. The matrix $\boldsymbol{Y}$ is considered as a collection of K points randomly chosen from the column space of $\boldsymbol{X}$, and hence the column space of $\boldsymbol{Y}$ is considered as a K-dimensional subspace of that of $\boldsymbol{X}$. Throughout this paper, we impose the following assumption on $\boldsymbol{Y}$.

Assumption 1 *$\boldsymbol{Y}$ is a column full-rank matrix.*

In the second step, one performs a QR decomposition on $\boldsymbol{Y}$ to obtain a matrix $\boldsymbol{Q} \in \mathbb{R}^{M\times K}$ and an upper triangular matrix $\boldsymbol{R} \in \mathbb{R}^{K\times K}$ that satisfy both $\boldsymbol{Y} = \boldsymbol{Q}\boldsymbol{R}$ and $\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{Q} = \boldsymbol{I}_K$, where $\boldsymbol{I}_K$ denotes the $K \times K$ identity matrix. Since the set of K columns of $\boldsymbol{Q}$ is an orthonormal basis for the column space of $\boldsymbol{Y}$, we have $\boldsymbol{X} \approx \boldsymbol{Q}\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{X} = \boldsymbol{Q}\boldsymbol{B}$, where $\boldsymbol{B} = \boldsymbol{Q}^{\mathrm{T}}\boldsymbol{X} \in \mathbb{R}^{K\times N}$ is called a surrogate matrix of $\boldsymbol{X}$.

In the third step, one transforms the optimization problem (1) into another with a lower dimension. Multiplying both sides of $\boldsymbol{X} \approx \boldsymbol{W}\boldsymbol{H}^{\mathrm{T}}$ by $\boldsymbol{Q}^{\mathrm{T}}$ from the left and letting $\tilde{\boldsymbol{W}} = \boldsymbol{Q}^{\mathrm{T}}\boldsymbol{W}$, we have $\boldsymbol{B} \approx \tilde{\boldsymbol{W}}\boldsymbol{H}^{\mathrm{T}}$. Since $\boldsymbol{W} \approx \boldsymbol{Q}\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{W} = \boldsymbol{Q}\tilde{\boldsymbol{W}}$ follows from $\boldsymbol{Q}\boldsymbol{Q}^{\mathrm{T}} \approx \boldsymbol{I}_M$, the nonnegativity constraint $\boldsymbol{W} \geq \boldsymbol{0}_{M\times K}$ is approximately equivalent to the linear inequality constraint $\boldsymbol{Q}\tilde{\boldsymbol{W}} \geq \boldsymbol{0}_{M\times K}$. Therefore, one obtains the following optimization problem in $\tilde{\boldsymbol{W}} \in \mathbb{R}^{K\times K}$ and $\boldsymbol{H} \in \mathbb{R}^{N\times K}$:

$$\begin{array}{ll} \text{minimize} & f_1(\tilde{\boldsymbol{W}}, \boldsymbol{H}) = \dfrac{1}{2}\left\|\boldsymbol{B} - \tilde{\boldsymbol{W}}\boldsymbol{H}^{\mathrm{T}}\right\|_{\mathrm{F}}^2 \\ \text{subject to} & \boldsymbol{Q}\tilde{\boldsymbol{W}} \geq \boldsymbol{0}_{M\times K},\ \boldsymbol{H} \geq \boldsymbol{0}_{N\times K}. \end{array} \tag{2}$$

Note that the number of variables in (2) is smaller than that in (1) because $K < M$. However, the feasible region of (2) is more complicated than that of (1).

In the last step, one finds an approximate local optimal solution $(\tilde{\boldsymbol{W}}^*, \boldsymbol{H}^*)$ of (2) in some way, and then returns $(\boldsymbol{Q}\tilde{\boldsymbol{W}}^*, \boldsymbol{H}^*)$ as an approximate local optimal solution of (1).

Erichson *et al*. [5] proposed an iterative algorithm based on the HALS algorithm for solving (2). However, their algorithm does not always return a solution that satisfies the linear inequality constraint $\boldsymbol{Q}\tilde{\boldsymbol{W}} \geq \boldsymbol{0}_{M\times K}$. The authors of the present paper proposed another iterative algorithm for solving (2) [6]. This algorithm converges to the set of stationary points, but is slow because only one variable is updated at a time.

3. Reformulation of Randomized NMF

Erichson *et al*. [5] transformed (1) into (2) using the change of variable $\tilde{\boldsymbol{W}} = \boldsymbol{Q}^{\mathrm{T}}\boldsymbol{W}$. In this approach, however, the nonnegativity constraint $\boldsymbol{W} \geq \boldsymbol{0}_{M\times K}$ in (1) is not exactly equivalent to the linear inequality constraint $\boldsymbol{Q}\tilde{\boldsymbol{W}} \geq \boldsymbol{0}_{M\times K}$ in (2) because $\boldsymbol{Q}\tilde{\boldsymbol{W}} = \boldsymbol{Q}\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{W}$ is not exactly equal to $\boldsymbol{W}$.

In order to avoid this ambiguity, we propose to transform (1) using the change of variable $\boldsymbol{W} = \boldsymbol{Q}\tilde{\boldsymbol{W}}$. In other words, we restrict the columns of $\boldsymbol{W}$ to those obtained by linear combination of the columns of $\boldsymbol{Q}$. Then, the problem (1) is transformed into the optimization problem:

$$\begin{array}{ll} \text{minimize} & f_2(\tilde{\boldsymbol{W}}, \boldsymbol{H}) = \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{Q}\tilde{\boldsymbol{W}}\boldsymbol{H}^{\mathrm{T}}\|_{\mathrm{F}}^2 \\ \text{subject to} & \boldsymbol{Q}\tilde{\boldsymbol{W}} \geq \boldsymbol{0}_{M\times K},\ \boldsymbol{H} \geq \boldsymbol{0}_{N\times K}. \end{array} \tag{3}$$

An advantage of our approach is that, for any feasible solution $(\tilde{\boldsymbol{W}}, \boldsymbol{H})$ of (3), we can immediately obtain a feasible solution of (1), which is given by $(\boldsymbol{Q}\tilde{\boldsymbol{W}}, \boldsymbol{H})$.

The problems (2) and (3) have the same constraints but different objective functions. However, these two problems are equivalent to each other as shown in the next theorem.

Theorem 1 The objective functions f_1 in (2) and f_2 in (3) differ only by a constant.

Proof: Let us define an orthonormal matrix $\boldsymbol{P} \in \mathbb{R}^{M\times M}$ and a matrix $\boldsymbol{V} \in \mathbb{R}^{M\times K}$ by

$$\boldsymbol{P} = \begin{pmatrix} \boldsymbol{Q}\ \bar{\boldsymbol{Q}} \end{pmatrix}, \quad \boldsymbol{V} = \begin{pmatrix} \tilde{\boldsymbol{W}} \\ \boldsymbol{0} \end{pmatrix}$$

where $\bar{\boldsymbol{Q}}$ is an $M \times (M - K)$ matrix satisfying $\bar{\boldsymbol{Q}}^{\mathrm{T}}\bar{\boldsymbol{Q}} = \boldsymbol{I}$ and $\boldsymbol{Q}^{\mathrm{T}}\bar{\boldsymbol{Q}} = \boldsymbol{0}$, that is, the set of $M - K$ columns of $\bar{\boldsymbol{Q}}$ is an orthonormal basis for the orthogonal complement of the column space of $\boldsymbol{Q}$. Using the equalities $\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P} = \boldsymbol{P}\boldsymbol{P}^{\mathrm{T}} = \boldsymbol{I}$, $\boldsymbol{P}\boldsymbol{V} = \boldsymbol{Q}\tilde{\boldsymbol{W}}$ and $\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V} = \tilde{\boldsymbol{W}}^{\mathrm{T}}\tilde{\boldsymbol{W}}$, we have

$$\begin{aligned} f_1(\tilde{\boldsymbol{W}}, \boldsymbol{H}) &= \frac{1}{2}\left\|\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{X} - \tilde{\boldsymbol{W}}\boldsymbol{H}^{\mathrm{T}}\right\|_{\mathrm{F}}^2 \\ &= \frac{1}{2}\left\|\begin{pmatrix} \boldsymbol{Q}^{\mathrm{T}} \\ \bar{\boldsymbol{Q}}^{\mathrm{T}} \end{pmatrix}\boldsymbol{X} - \begin{pmatrix} \tilde{\boldsymbol{W}} \\ \boldsymbol{0} \end{pmatrix}\boldsymbol{H}^{\mathrm{T}}\right\|_{\mathrm{F}}^2 - \frac{1}{2}\left\|\bar{\boldsymbol{Q}}^{\mathrm{T}}\boldsymbol{X}\right\|_{\mathrm{F}}^2 \\ &= \frac{1}{2}\left\|\boldsymbol{P}^{\mathrm{T}}\boldsymbol{X} - \boldsymbol{V}\boldsymbol{H}^{\mathrm{T}}\right\|_{\mathrm{F}}^2 - \frac{1}{2}\left\|\bar{\boldsymbol{Q}}^{\mathrm{T}}\boldsymbol{X}\right\|_{\mathrm{F}}^2 \\ &= \frac{1}{2}\sum_{n=1}^{N}\left\|\boldsymbol{P}^{\mathrm{T}}\boldsymbol{x}_n - \boldsymbol{V}(\boldsymbol{H}^{\mathrm{T}})_n\right\|_2^2 - \frac{1}{2}\left\|\bar{\boldsymbol{Q}}^{\mathrm{T}}\boldsymbol{X}\right\|_{\mathrm{F}}^2 \\ &= \frac{1}{2}\sum_{n=1}^{N}\Big(\boldsymbol{x}_n^{\mathrm{T}}\boldsymbol{x}_n - 2\boldsymbol{x}_n^{\mathrm{T}}\boldsymbol{Q}\tilde{\boldsymbol{W}}(\boldsymbol{H}^{\mathrm{T}})_n \\ &\quad - ((\boldsymbol{H}^{\mathrm{T}})_n)^{\mathrm{T}}\tilde{\boldsymbol{W}}^{\mathrm{T}}\tilde{\boldsymbol{W}}(\boldsymbol{H}^{\mathrm{T}})_n\Big) - \frac{1}{2}\left\|\bar{\boldsymbol{Q}}^{\mathrm{T}}\boldsymbol{X}\right\|_{\mathrm{F}}^2 \\ &= \frac{1}{2}\sum_{n=1}^{N}\left\|\boldsymbol{x}_n - \boldsymbol{Q}\tilde{\boldsymbol{W}}(\boldsymbol{H}^{\mathrm{T}})_n\right\|_2^2 - \frac{1}{2}\left\|\bar{\boldsymbol{Q}}^{\mathrm{T}}\boldsymbol{X}\right\|_{\mathrm{F}}^2 \\ &= f_2(\tilde{\boldsymbol{W}}, \boldsymbol{H}) - \frac{1}{2}\left\|\bar{\boldsymbol{Q}}^{\mathrm{T}}\boldsymbol{X}\right\|_{\mathrm{F}}^2. \end{aligned}$$

Since the second term of the last formula is a constant, the functions f_1 and f_2 differ only by a constant. □

In the first formulation of the randomized NMF [5], Erichson *et al.* assumed that the number of columns of $\boldsymbol{\Omega}$ is equal to the target rank K. However, they also considered the case where the number of columns of $\boldsymbol{\Omega}$ is greater than K. We thus hereafter assume that $\boldsymbol{\Omega}$ is an $N \times L$ random matrix, where L is an integer satisfying $K \leq L \leq M - 1$. In this case, $\boldsymbol{Q}$ is an $M \times L$ matrix and $\tilde{\boldsymbol{W}}$ is an $L \times K$ matrix.

4. Proposed Algorithm

The optimization problem (3) is equivalent to (2) as shown in Theorem 1. Also, the objective function of (2) is simpler than that of (3). Therefore, we focus our attention on (2) in this paper, and design an algorithm for finding a stationary point.

It was reported by Erichson *et al*. [5] that $\boldsymbol{\Omega}$ based on the uniform distribution on $[0, 1]$ gives better results than the standard normal distribution. We thus impose the following assumptions on the matrices $\boldsymbol{\Omega}$ and $\boldsymbol{Q}$.

Assumption 2 The entries of $\boldsymbol{\Omega} \in \mathbb{R}^{N\times L}$ are drawn i.i.d. from the uniform distribution on $[0, 1]$. All entries in the first column of $\boldsymbol{Q} \in \mathbb{R}^{M\times L}$ are nonnegative, and there exists at least one nonzero entry in each row of $\boldsymbol{Q}$.

Note that $\boldsymbol{X\Omega}$ is nonnegative when $\boldsymbol{\Omega}$ is nonnegative. So we can assume without loss of generality that the first column of $\boldsymbol{Q}$ is nonnegative. The last condition in Assumption 2 is also very mild, and usually satisfied.

We now present an overview of our iterative algorithm for finding a stationary point of the optimization problem (2). Given an initial feasible solution $(\tilde{\boldsymbol{W}}, \boldsymbol{H})$ such that

$$\tilde{w}_{lk}\begin{cases}\geq 0, & \text{if } l = 1,\\ = 0, & \text{if } l \in \{2, 3, \ldots, L\},\end{cases} \tag{4}$$

our algorithm updates the $2K$ columns of $\tilde{\boldsymbol{W}}$ and $\boldsymbol{H}$ one by one in the order $\boldsymbol{h}_1 \to \tilde{\boldsymbol{w}}_1 \to \boldsymbol{h}_2 \to \tilde{\boldsymbol{w}}_2 \to \cdots \to \boldsymbol{h}_K \to \tilde{\boldsymbol{w}}_K$, where $\tilde{\boldsymbol{w}}_k$ and $\boldsymbol{h}_k$ denotes the k-th column of $\tilde{\boldsymbol{W}}$ and $\boldsymbol{H}$, respectively. Note that the initial $\tilde{\boldsymbol{W}}$ given by (4) satisfies $\boldsymbol{Q}\tilde{\boldsymbol{W}} \geq \boldsymbol{0}$ because of Assumption 2.

The update rule for $(\tilde{\boldsymbol{W}}, \boldsymbol{H})$ in our algorithm is shown in Algorithm 1. In Step 3 of Algorithm 1, $\boldsymbol{h}_k$ is updated in the same way as the one presented in the literature [2, 4], where δ is a positive constant and $[\boldsymbol{t}]_+$ represents the vector obtained from $\boldsymbol{t}$ by replacing all negative entries with zeros. In Step 4, $\boldsymbol{h}_k$ is normalized to guarantee the boundedness of the sequence of solutions generated. Note that this normalization process does not change the value of $\tilde{\boldsymbol{w}}_k\boldsymbol{h}_k^{\mathrm{T}}$ and hence the value of the objective function $f_1(\tilde{\boldsymbol{W}}, \boldsymbol{H})$. In Steps 5–16, the system of equations:

$$\begin{aligned}\tilde{\boldsymbol{w}}_k - \boldsymbol{S}_k\boldsymbol{h}_k - \boldsymbol{Q}^{\mathrm{T}}\boldsymbol{\lambda}_k &= \boldsymbol{0}_{L\times 1},\\ \boldsymbol{\lambda}_k \odot (\boldsymbol{Q}\tilde{\boldsymbol{w}}_k) &= \boldsymbol{0}_{M\times 1},\\ \boldsymbol{\lambda}_k &\geq \boldsymbol{0}_{M\times 1},\end{aligned}$$

which is the Karush-Kuhn-Tucker condition for the optimization problem:

$$\begin{array}{ll}\text{minimize} & \frac{1}{2}\|\tilde{\boldsymbol{w}}_k\|_2^2 - (\boldsymbol{S}_k\boldsymbol{h}_k)^{\mathrm{T}}\tilde{\boldsymbol{w}}_k\\ \text{subject to} & \boldsymbol{Q}\tilde{\boldsymbol{w}}_k \geq \boldsymbol{0}_{M\times K},\end{array} \tag{5}$$

is solved approximately for $\tilde{\boldsymbol{w}}_k$ and $\boldsymbol{\lambda}_k$.

The global convergence of the proposed algorithm is shown as follows. The proof is omitted due to lack of space.

Algorithm 1 Update Rule for Randomized NMF

Input: $\boldsymbol{X} \in \mathbb{R}_+^{M\times N}$, $\boldsymbol{Q} \in \mathbb{R}^{M\times L}$, $\tilde{\boldsymbol{W}} \in \mathbb{R}^{L\times K}$, $\boldsymbol{H} \in \mathbb{R}_+^{N\times K}$, $\boldsymbol{\Lambda} \in \mathbb{R}_+^{M\times K}$, $\delta, \tau_g \in \mathbb{R}_{++}$, $J \in \mathbb{Z}_{++}$.
Output: Updated $\tilde{\boldsymbol{W}}$, $\boldsymbol{H}$ and $\boldsymbol{\Lambda}$.

1: Set $\boldsymbol{E} \leftarrow \boldsymbol{Q}^{\mathrm{T}}\boldsymbol{X} - \tilde{\boldsymbol{W}}\boldsymbol{H}^{\mathrm{T}}$ and $k \leftarrow 1$.
2: Set $\boldsymbol{S}_k \leftarrow \boldsymbol{E} + \tilde{\boldsymbol{w}}_k\boldsymbol{h}_k^{\mathrm{T}}$.
3: Set $\boldsymbol{h}_k \leftarrow [(\boldsymbol{S}_k^{\mathrm{T}}\tilde{\boldsymbol{w}}_k + \delta\boldsymbol{h}_k)/(\|\tilde{\boldsymbol{w}}_k\|_2^2 + \delta)]_+$.
4: If $\boldsymbol{h}_k \neq \boldsymbol{0}_{N\times 1}$ then $\tilde{\boldsymbol{w}}_k \leftarrow \tilde{\boldsymbol{w}}_k\|\boldsymbol{h}_k\|_2$ and $\boldsymbol{h}_k \leftarrow \boldsymbol{h}_k/\|\boldsymbol{h}_k\|_2$. Otherwise set $\tilde{\boldsymbol{w}}_k \leftarrow \boldsymbol{0}_{K\times 1}$ and $\boldsymbol{h}_k \leftarrow \boldsymbol{u}_k$, where $\boldsymbol{u}_k \in \mathbb{R}_+^N$ is any nonnegative unit vector.
5: Set $\tilde{\boldsymbol{w}}_k^{\min} \leftarrow \boldsymbol{S}_k\boldsymbol{h}_k$.
6: Set $I_{\mathrm{pos}} \leftarrow \{m \in \{1, 2, \ldots, M\} \mid \lambda_{mk} > 0\}$ and $\boldsymbol{P} \leftarrow \boldsymbol{0}_{L\times L}$.
7: For each $m \in I_{\mathrm{pos}}$, set $\boldsymbol{v}_m \leftarrow (\boldsymbol{Q}^{\mathrm{T}})_m - \boldsymbol{P}(\boldsymbol{Q}^{\mathrm{T}})_m$, and then set $\boldsymbol{P} \leftarrow \boldsymbol{P} + \boldsymbol{v}_m\boldsymbol{v}_m^{\mathrm{T}}/\|\boldsymbol{v}_m\|_2^2$ if $\boldsymbol{v}_m \neq \boldsymbol{0}_{L\times 1}$.
8: Set $\tilde{\boldsymbol{w}}_k^* \leftarrow \tilde{\boldsymbol{w}}_k^{\min} - \boldsymbol{P}\tilde{\boldsymbol{w}}_k^{\min}$.
9: If $\boldsymbol{Q}\tilde{\boldsymbol{w}}_k^* \geq \boldsymbol{0}_{M\times 1}$ then set $\tilde{\boldsymbol{w}}_k \leftarrow \tilde{\boldsymbol{w}}_k^*$ and go to Step 12. Otherwise set $\theta_{\max} \leftarrow \max\{\theta \in [0, 1] \mid \boldsymbol{Q}(\tilde{\boldsymbol{w}}_k + \theta(\tilde{\boldsymbol{w}}_k^* - \tilde{\boldsymbol{w}}_k)) \geq \boldsymbol{0}_{M\times 1}\}$, $\tilde{\boldsymbol{w}}_k \leftarrow \tilde{\boldsymbol{w}}_k + \theta_{\max}(\tilde{\boldsymbol{w}}_k^* - \tilde{\boldsymbol{w}}_k)$ and $I_{\mathrm{new}} \leftarrow \{m \in \{1, 2, \ldots, M\} \mid (\boldsymbol{Q}\tilde{\boldsymbol{w}}_k)_m = 0\} \setminus I_{\mathrm{pos}}$.
10: For each $m \in I_{\mathrm{new}}$, set $\boldsymbol{v}_m \leftarrow (\boldsymbol{Q}^{\mathrm{T}})_m - \boldsymbol{P}(\boldsymbol{Q}^{\mathrm{T}})_m$ and then set $\boldsymbol{P} \leftarrow \boldsymbol{P} + \boldsymbol{v}_m\boldsymbol{v}_m^{\mathrm{T}}/\|\boldsymbol{v}_m\|_2^2$ if $\boldsymbol{v}_m \neq \boldsymbol{0}_{L\times 1}$.
11: Set $\tilde{\boldsymbol{w}}_k^* \leftarrow \tilde{\boldsymbol{w}}_k^{\min} - \boldsymbol{P}\tilde{\boldsymbol{w}}_k^{\min}$ and return to Step 9.
12: Set $I_{\mathrm{act}} \leftarrow \{m \in \{1, 2, \ldots, M\} \mid (\boldsymbol{Q}\tilde{\boldsymbol{w}}_k)_m = 0\}$.
13: Set $j \leftarrow 1$.
14: Update the values of λ_{mk} $(m \in I_{\mathrm{act}})$ one by one as

$$\lambda_{mk} \leftarrow \left[\frac{\left(-\tilde{\boldsymbol{w}}_k^{\min} - \sum_{m'\in I_{\mathrm{act}}\setminus\{m\}} \lambda_{m'k}(\boldsymbol{Q}^{\mathrm{T}})_{m'}\right)^{\mathrm{T}}(\boldsymbol{Q}^{\mathrm{T}})_m}{\|(\boldsymbol{Q}^{\mathrm{T}})_m\|_2^2}\right]_+.$$

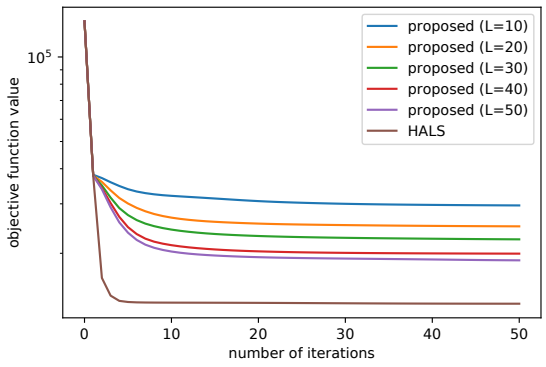
15: If $j < J$ then set $j \leftarrow j + 1$ and return to Step 14. Otherwise return to Step 6.
16: If $\frac{1}{2}\|\tilde{\boldsymbol{w}}_k - \tilde{\boldsymbol{w}}_k^{\min} - \boldsymbol{Q}^{\mathrm{T}}\boldsymbol{\lambda}_k\|_2^2 \leq \tau_g$ then go to Step 17.
17: Set $\boldsymbol{E} \leftarrow \boldsymbol{S}_k - \tilde{\boldsymbol{w}}_k\boldsymbol{h}_k^{\mathrm{T}}$.
18: If $k = K$ then return $\tilde{\boldsymbol{W}}$, $\boldsymbol{H}$, $\boldsymbol{\Lambda}$ and stop. Otherwise set $k \leftarrow k + 1$ and return to Step 2.

Theorem 2 Suppose that τ_g is sufficiently small. Then, for any initial solution $(\tilde{\boldsymbol{W}}^{(0)}, \boldsymbol{H}^{(0)})$, the sequence of solutions $\{(\tilde{\boldsymbol{W}}^{(t)}, \boldsymbol{H}^{(t)})\}_{t=0}^{\infty}$ generated by Algorithm 1 has at least one convergent subsequence, and the limit of any convergent subsequence is a stationary point of (2).

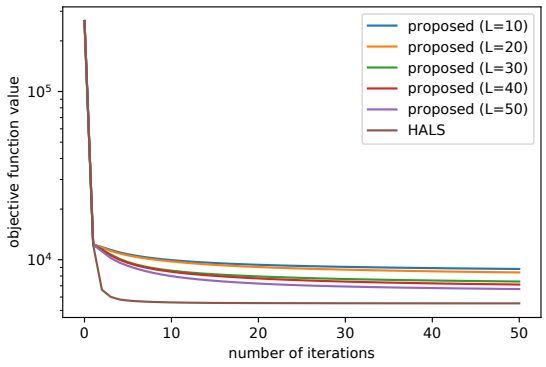
5. Experiments

In order to evaluate the validity and effectiveness of the proposed algorithm, the authors conducted experiments using two image datasets COIL-20 [7] and Olivetti*. From the COIL-20 dataset, a 1024×1440 matrix $\boldsymbol{X}$ with values between 0 and 1 was constructed. From the Olivetti dataset, a 4096×400 matrix $\boldsymbol{X}$ with values between 0 and 1 was constructed. For these two nonnegative matrices, the target rank K was set to 10 and L, the number of columns of

*https://scikit-learn.org/0.19/datasets/olivetti_faces.html



(a)



(b)

Figure 1: Evolution of the value of $f(\boldsymbol{W}, \boldsymbol{H})$: (a) COIL-20 and (b) Olivetti.

$\boldsymbol{\Omega}$, was set to 10, 20, 30, 40 and 50. The random matrix $\boldsymbol{\Omega}$ is generated using pseudo-random numbers from the uniform distribution on $[0, 1]$. The (l, k)-th entry of $\tilde{\boldsymbol{W}}$ was initialized to 1 if $l = 1$ and 0 otherwise so that the condition (4) is satisfied. $\boldsymbol{H}$ was initialized using pseudo-random numbers from the uniform distribution on $[0, 0.1]$, and $\boldsymbol{\Lambda}$ was initialized to $\boldsymbol{0}_{M\times K}$. The constants were set as follows: $\delta = 10^{-12}$, $\tau_g = 10^{-2}$ and $J = 20$. The proposed algorithm and the HALS algorithm [4] were implemented in Python 3.9.7 and executed on a computer with Intel Core i5-9500 CPU, 8GB RAM and Windows 10 Pro.

Figure 1 shows the evolution of the objective function $f(\boldsymbol{W}, \boldsymbol{H})$ of (1) for the first 50 iterations. Since $\boldsymbol{W}$ is restricted to those given by $\boldsymbol{W} = \boldsymbol{Q}\tilde{\boldsymbol{W}}$ in the proposed algorithm, the value of $f(\boldsymbol{W}, \boldsymbol{H})$ is higher than the HALS algorithm. It is easily seen from the figures that the value of $f(\boldsymbol{W}, \boldsymbol{H})$ decreases monotonically in all cases, and that the larger the value of L, the smaller the value of $f(\boldsymbol{W}, \boldsymbol{H})$. Table 1 shows the computation time of the proposed and HALS algorithms for the first 50 iterations. The proposed algorithm requires shorter computation time than the HALS algorithm when $L \in \{10, 20, 30, 40\}$, but there is no significant difference when $L = 50$.

Table 1: Computation time for the first 50 iterations.

Algorithm	COIL-20 (sec)	Olivetti (sec)
Proposed ($L = 10$)	3.65	3.65
Proposed ($L = 20$)	6.75	6.29
Proposed ($L = 30$)	7.56	8.60
Proposed ($L = 40$)	8.04	9.86
Proposed ($L = 50$)	10.84	14.73
HALS [4]	12.16	13.40

6. Conclusions

We have reformulated randomized NMF using the change of variable $\boldsymbol{W} = \boldsymbol{Q}\tilde{\boldsymbol{W}}$, and proposed a new iterative algorithm for solving the optimization problem of randomized NMF. We have also shown through experiments using two real datasets that the proposed algorithm is effective in terms of computation time when L is small. However, when L is large, the proposed algorithm has no significant advantage over the HALS algorithm. Developing faster algorithms for randomized NMF is a future challenge.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number JP21H03510.

References

[1] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[2] N. Gillis, *Nonnegative Matrix Factorization*, SIAM, 2020.

[3] A. Cichocki, R. Zdunek and S. I. Amari, "Hierarchical ALS algorithms for nonnegative matrix and 3D tensor factorization," *Proc. ICA2007*, pp. 169–176, 2007.

[4] T. Sano, T. Migita and N. Takahashi, "A novel update rule of HALS algorithm for nonnegative matrix factorization and Zangwill's global convergence," *Journal of Global Optimization*, vol. 84, no. 3, pp.755–781, 2022.

[5] N. B. Erichson, A. Mendible, S. Wihiborn and J. N. Kutz, "Randomized nonnegative matrix factorization," *Pattern Recognition Letters*, vol. 104, pp. 1–7, 2018.

[6] T. Masuda, T. Migita and N. Takahashi, "An algorithm for randomized nonnegative matrix factorization and its global convergence," *Proc. IEEE SSCI2021*, pp.1–7, 2021.

[7] S. Nene, S. Nayar and H. Murase, "Columbia object image library (COIL-20)," Technical Report CUCS-005-96, 1996.