LETTER Communicated by Arindam Banerjee

Multilinear Common Component Analysis via Kronecker Product Representation

Kohei Yoshikawa
yoshikawa.kohei615@gmail.com
Shuichi Kawano
skawano@ai.lab.uec.ac.jp
Graduate School of Informatics and Engineering, The University of Electro-Communications, Chofu-shi, Tokyo 182-8585, Japan

We consider the problem of extracting a common structure from multiple tensor data sets. For this purpose, we propose multilinear common component analysis (MCCA) based on Kronecker products of mode-wise covariance matrices. MCCA constructs a common basis represented by linear combinations of the original variables that lose little information of the multiple tensor data sets. We also develop an estimation algorithm for MCCA that guarantees mode-wise global convergence. Numerical studies are conducted to show the effectiveness of MCCA.

1 Introduction

Various statistical methodologies for extracting useful information from a large amount of data have been studied over the decades since the appearance of big data. In the present era, it is important to discover a common structure of multiple data sets. In an early study, Flury (1984) focused on the structure of the covariance matrices of multiple data sets and discussed the heterogeneity of the structure. The author reported that population covariance matrices differ among multiple data sets in practical applications. Many methodologies have been developed for treating the heterogeneity between covariance matrices of multiple data sets (see, Flury, 1986, 1988; Flury & Gautschi, 1986; Pourahmadi, Daniels, & Park, 2007; Wang, Banerjee, & Boley, 2011; Park & Konishi, 2020).

Among such methodologies, common component analysis (CCA; Wang et al., 2011) is an effective tool for statistics. The central idea of CCA is to reduce the number of dimensions of data while losing as little information of the multiple data sets as possible. To reduce the number of dimensions, CCA reconstructs the data with a few new variables that are linear combinations of the original variables. For considering the heterogeneity between covariance matrices of multiple data sets, CCA assumes that there is a different covariance matrix for each data set. There have been many papers on various statistical methodologies using multiple covariance matrices:

Neural Computation **33, 2853–2880** (2021)
https://doi.org/10.1162/neco_a_01425

discriminant analysis (Bensmail & Celeux, 1996), spectral decomposition (Boik, 2002), and a likelihood ratio test for multiple covariance matrices (Manly & Rayner, 1987). It should be noted that principal component analysis (PCA) (Pearson, 1901; Jolliffe, 2002) is a technique similar to CCA. In fact, CCA is a generalization of PCA; PCA can only be applied to one data set, whereas CCA can be applied to multiple data sets.

Meanwhile, in various fields of research, including machine learning and computer vision, the main interest has been in tensor data, which has a multidimensional array structure. In order to apply the conventional statistical methodologies, such as PCA, to tensor data, a simple approach is to first transform the tensor data into vector data and then apply the methodology. However, such an approach causes the following problems:

1. In losing the tensor structure of the data, the approach ignores the higher-order inherent relationships of the original tensor data.
2. Transforming tensor data to vector data substantially increases the number of features. It also has a high computational cost.

To overcome these problems, statistical methodologies for tensor data analyses have been proposed that take the tensor structure of the data into consideration. Such methods enable us to accurately extract higher-order inherent relationships in a tensor data set. In particular, many existing statistical methodologies have been extended for tensor data, for example, multilinear principal component analysis (MPCA) (Lu et al., 2008) and sparse PCA for tensor data analysis (Allen, 2012; Wang, Sun, Chen, Pang, & Zhou, 2012; Lai, Xu, Chen, Yang, & Zhang, 2014), as well as others (see Carroll & Chang, 1970; Harshman, 1970; Kiers, 2000; Badeau & Boyer, 2008; Kolda & Bader, 2009).

In this letter, we extend CCA to tensor data analysis, proposing multilinear common component analysis (MCCA). MCCA discovers the common structure of multiple data sets of tensor data while losing as little of the information of the data sets as possible. To identify the common structure, we estimate a common basis constructed as linear combinations of the original variables. For estimating the common basis, we develop a new estimation algorithm based on the idea of CCA. In developing the estimation algorithm, two issues must be addressed: the convergence properties of the algorithm and its computational cost. To determine the convergence properties, we investigate first the relationship between the initial values of the parameters and global optimal solution and then the monotonic convergence of the estimation algorithm. These analyses reveal that our proposed algorithm guarantees convergence of the mode-wise global optimal solution under some conditions. To analyze the computational efficacy, we calculate the computational cost of our proposed algorithm.

The rest of the letter is organized as follows. In section 2, we review the formulation and the minimization problem of CCA. In section 3, we formulate the MCCA model by constructing the covariance matrices of tensor data, based on a Kronecker product representation. Then we formulate the estimation algorithm for MCCA in section 4. In section 5, we present the theoretical properties for our proposed algorithm and analyze the computational cost. The efficacy of the MCCA is demonstrated through numerical experiments in section 6. Concluding remarks are presented in section 7. Technical proofs are provided in the appendixes. Our implementation of MCCA and supplementary materials are available at https://github.com/yoshikawa-kohei/MCCA.

2 Common Component Analysis

Suppose that we obtain data matrices $\mathbf{X}_{(g)} = [\mathbf{x}_{(g)1}, \dots \mathbf{x}_{(g)N_g}]^\top \in \mathbb{R}^{N_g \times P}$ with N_g observations and P variables for $g = 1, \dots, G$, where $\mathbf{x}_{(g)i}$ is the P-dimensional vector corresponding to the ith row of $\mathbf{X}_{(g)}$ and G is the number of data sets. Then the sample covariance matrix in group g is

$$\mathbf{S}_{(g)} = \frac{1}{N_g} \sum_{i=1}^{N_g} \left(\mathbf{x}_{(g)i} - \bar{\mathbf{x}}_{(g)} \right) \left(\mathbf{x}_{(g)i} - \bar{\mathbf{x}}_{(g)} \right)^\top, \quad g = 1, \dots, G, \tag{2.1}$$

where $\mathbf{S}_{(g)} \in \mathbb{S}_{+}^{P}$, in which $\mathbb{S}_{+}^{P}$ is a set of symmetric positive-definite matrices of size $P \times P$, and $\bar{\mathbf{x}}_{(g)} = \frac{1}{N_g} \sum_{i=1}^{N_g} \mathbf{x}_{(g)i}$ is a P-dimensional mean vector in group g.

The main idea of the CCA model is to find the common structure of multiple data sets by projecting the data onto a common lower-dimensional space with the same basis as the data sets. Wang et al. (2011) assumed that the covariance matrices $\mathbf{S}_{(g)}$ for $g = 1, \dots, G$ can be decomposed to a product of latent covariance matrices and an orthogonal matrix for the linear transformation as

$$\mathbf{S}_{(g)} = \mathbf{V} \boldsymbol{\Lambda}_{(g)} \mathbf{V}^\top + \mathbf{E}_{(g)}, \quad \text{s.t.} \quad \mathbf{V}^\top \mathbf{V} = \mathbf{I}_R, \tag{2.2}$$

where $\boldsymbol{\Lambda}_{(g)} \in \mathbb{S}_{+}^{R}$ is the latent covariance matrix in group g, $\mathbf{V} \in \mathbb{R}^{P \times R}$ is an orthogonal matrix for the linear transformation, $\mathbf{E}_{(g)} \in \mathbb{S}_{+}^{P}$ is the error matrix in group g, and $\mathbf{I}_R$ is the identity matrix of size $R \times R$. $\mathbf{E}_{(g)}$ consists of the sum of outer products for independent random vectors $\sum_{i=1}^{N_g} \mathbf{e}_{(g)i} \mathbf{e}_{(g)i}^\top$ with mean $\mathrm{E}\left[\mathbf{e}_{(g)i}\right] = \mathbf{0}$ and covariance matrix $\mathrm{Cov}\left[\mathbf{e}_{(g)i}\right]$ $(> \boldsymbol{O})$ $(i = 1, 2, \dots, N_g)$. $\mathbf{V}$ determines the R-dimensional common subspace of the multiple data sets. In particular, by assuming $R < P$, the CCA can discover the latent structures of the data sets. Wang et al. (2011) referred to the model, equation 2.2, as *common component analysis* (CCA).

The parameters $\mathbf{V}$ and $\boldsymbol{\Lambda}_{(g)}$ $(g = 1, \dots, G)$ are estimated by solving the minimization problem,

$$\min_{\substack{\mathbf{V},\mathbf{\Lambda}_{(g)}\\ g=1,\dots,G}} \sum_{g=1}^{G} \|\mathbf{S}_{(g)} - \mathbf{V}\mathbf{\Lambda}_{(g)}\mathbf{V}^\top\|_F^2, \quad \text{s.t.} \quad \mathbf{V}^\top\mathbf{V} = \mathbf{I}_R, \tag{2.3}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. The estimator of latent covariance matrices $\mathbf{\Lambda}_{(g)}$ for $g = 1, \dots, G$ can be obtained by solving the minimization problem as $\hat{\mathbf{\Lambda}}_{(g)} = \mathbf{V}^\top\mathbf{S}_{(g)}\mathbf{V}$. By using the estimated value $\hat{\mathbf{\Lambda}}_{(g)}$, the minimization problem can be reformulated as the following maximization problem:

$$\max_{\mathbf{V}} \operatorname{tr}\left\{\mathbf{V}^\top \sum_{g=1}^{G} \left(\mathbf{S}_{(g)}\mathbf{V}\mathbf{V}^\top\mathbf{S}_{(g)}\right)\mathbf{V}\right\}, \quad \text{s.t.} \quad \mathbf{V}^\top\mathbf{V} = \mathbf{I}_R, \tag{2.4}$$

where $\operatorname{tr}(\cdot)$ denotes the trace of a matrix. A crucial issue for solving the maximization problem 2.4 is the nonconvexity. Certainly the maximization problem is nonconvex since the problem is defined on a set of orthogonal matrices, which is a nonconvex set. Generally it is difficult to find the global optimal solution in nonconvex optimization problems. To overcome this drawback, Wang et al. (2011) proposed an estimation algorithm in which the estimated parameters are guaranteed to constitute the global optimal solution under some conditions.

3 Multilinear Common Component Analysis

In this section, we introduce a mathematical formulation of the MCCA, an extension of the CCA in terms of tensor data analysis. Moreover, we formulate an optimization problem of MCCA and investigate its convergence properties.

Suppose that we independently obtain Mth order tensor data $\mathcal{X}_{(g)i} \in \mathbb{R}^{P_1 \times P_2 \times \ldots \times P_M}$ for $i = 1, \ldots N_g$. We set the data sets of the tensors $\mathcal{X}_{(g)} = [\mathcal{X}_{(g)1}, \mathcal{X}_{(g)2}, \dots, \mathcal{X}_{(g)N_g}] \in \mathbb{R}^{P_1 \times P_2 \times \cdots \times P_M \times N_g}$ for $g = 1, \dots, G$, where G is the number of data sets. Then the sample covariance matrix in group g for the tensor data set is defined by

$$\mathbf{S}^*_{(g)} := \mathbf{S}^{(1)}_{(g)} \otimes \mathbf{S}^{(2)}_{(g)} \otimes \cdots \otimes \mathbf{S}^{(M)}_{(g)}, \tag{3.1}$$

where $\mathbf{S}^*_{(g)} \in \mathbb{S}^P_+$, in which $P = \prod_{k=1}^{M} P_k$, $\otimes$ denotes the Kronecker product operator, and $\mathbf{S}^{(k)}_{(g)} \in \mathbb{S}^{P_k}_+$ is the sample covariance matrix for kth mode in group g defined by

$$\mathbf{S}^{(k)}_{(g)} := \frac{1}{N_g \prod_{j \neq k} P_j} \sum_{i=1}^{N_g} \left(\mathbf{X}^{(k)}_{(g)i} - \bar{\mathbf{X}}^{(k)}_{(g)}\right)\left(\mathbf{X}^{(k)}_{(g)i} - \bar{\mathbf{X}}^{(k)}_{(g)}\right)^\top. \tag{3.2}$$

Here, $\mathbf{X}_{(g)i}^{(k)} \in \mathbb{R}^{P_k \times (\prod_{j \neq k} P_j)}$ is the mode-k unfolded matrix of $\mathcal{X}_{(g)i}$, and $\bar{\mathbf{X}}_{(g)}^{(k)} \in \mathbb{R}^{P_k \times (\prod_{j \neq k} P_j)}$ is the mode-k unfolded matrix of $\bar{\mathcal{X}}_{(g)} = \frac{1}{N_g} \sum_{i=1}^{N_g} \mathcal{X}_{(g)i}$. Note that the mode-$k$ unfolding from an Mth order tensor $\mathcal{X} \in \mathbb{R}^{P_1 \times P_2 \times \cdots \times P_M}$ to a matrix $\mathbf{X}^{(k)} \in \mathbb{R}^{P_k \times (\prod_{j \neq k} P_j)}$ means that the tensor element $(p_1, p_2, \ldots, p_M)$ maps to matrix element (p_k, l), where $l = 1 + \sum_{t=1, t \neq k}^{M} (p_t - 1) L_t$ with $L_t = \prod_{m=1, m \neq k}^{t-1} P_m$, in which $p_1, p_2, \ldots, p_M$ denote the indices of the Mth order tensor $\mathcal{X}$. For a more detailed description of tensor operations, see Kolda and Bader (2009). A representation of the tensor covariance matrix by Kronecker products is often used (Kermoal, Schumacher, Pedersen, Mogensen, & Frederiksen, 2002; Yu et al., 2004; Werner, Jansson, & Stoica, 2008).

To formulate CCA in terms of tensor data analysis, we consider CCA for the kth mode covariance matrix in group g as follows,

$$\mathbf{S}_{(g)}^{(k)} = \mathbf{V}^{(k)} \boldsymbol{\Lambda}_{(g)}^{(k)} \mathbf{V}^{(k)\top} + \mathbf{E}_{(g)}^{(k)}, \quad \text{s.t.} \quad \mathbf{V}^{(k)\top} \mathbf{V}^{(k)} = \mathbf{I}_{R_k}, \tag{3.3}$$

where $\boldsymbol{\Lambda}_{(g)}^{(k)} \in \mathbb{S}_{+}^{R_k}$ is the latent kth mode covariance matrix in group g, $\mathbf{V}^{(k)} \in \mathbb{R}^{P_k \times R_k}$ is an orthogonal matrix for the linear transformation, and $\mathbf{E}_{(g)}^{(k)} \in \mathbb{S}_{+}^{P_k}$ is the error matrix in group g. $\mathbf{E}_{(g)}^{(k)}$ consists of the sum of outer products for independent random vectors $\sum_{i=1}^{N_g} \mathbf{e}_{(g)i}^{(k)} \mathbf{e}_{(g)i}^{(k)\top}$ with mean $\mathrm{E}\left[\mathbf{e}_{(g)i}^{(k)}\right] = \mathbf{0}$ and covariance matrix $\mathrm{Cov}\left[\mathbf{e}_{(g)i}^{(k)}\right]$ $(> \boldsymbol{O})$ $(i = 1, 2, \ldots, N_g)$. Since $\mathbf{S}_{(g)}^{*}$ can be decomposed to a Kronecker product of $\mathbf{S}_{(g)}^{(k)}$ for $k = 1, \ldots, M$ in formula 3.1, we obtain the following model,

$$\mathbf{S}_{(g)}^{*} = \mathbf{V}^{*} \boldsymbol{\Lambda}_{(g)}^{*} \mathbf{V}^{*\top} + \mathbf{E}_{(g)}^{*}, \quad \text{s.t.} \quad \mathbf{V}^{*\top} \mathbf{V}^{*} = \mathbf{I}_{R}, \tag{3.4}$$

where $R = \prod_{k=1}^{M} R_k$, $\mathbf{V}^{*} = \mathbf{V}^{(1)} \otimes \mathbf{V}^{(2)} \otimes \cdots \otimes \mathbf{V}^{(M)}$, $\boldsymbol{\Lambda}_{(g)}^{*} = \boldsymbol{\Lambda}_{(g)}^{(1)} \otimes \boldsymbol{\Lambda}_{(g)}^{(2)} \otimes \cdots \otimes \boldsymbol{\Lambda}_{(g)}^{(M)}$, and $\mathbf{E}_{(g)}^{*}$ is the error matrix in group g. We refer to this model as *multilinear common component analysis* (MCCA).

To find the R-dimensional common subspace between the multiple tensor data sets, MCCA determines $\mathbf{V}^{(1)}, \mathbf{V}^{(2)}, \ldots, \mathbf{V}^{(M)}$. As with CCA, we obtain the estimate of $\boldsymbol{\Lambda}_{(g)}^{*}$ for $g = 1, \ldots, G$ as $\hat{\boldsymbol{\Lambda}}_{(g)}^{*} = \mathbf{V}^{*\top} \mathbf{S}_{(g)}^{*} \mathbf{V}^{*}$. With respect to $\mathbf{V}^{*}$, we can obtain the estimate by solving the following maximization problem, which is similar to equation 2.4:

$$\max_{\mathbf{V}^{*}} \operatorname{tr} \left\{ \mathbf{V}^{*\top} \sum_{g=1}^{G} \left(\mathbf{S}_{(g)}^{*} \mathbf{V}^{*} \mathbf{V}^{*\top} \mathbf{S}_{(g)}^{*} \right) \mathbf{V}^{*} \right\}, \quad \text{s.t.} \quad \mathbf{V}^{*\top} \mathbf{V}^{*} = \mathbf{I}_{R}. \tag{3.5}$$

However, the number of parameters will be very large when we try to solve this problem directly, and thus results in a high computational cost.

Moreover, it may not be possible to discover the inherent relationships among the variables in each mode simply by solving problem 3.5.

To solve the maximization problem efficiently and identify the inherent relationships, the maximization problem 3.5 can be decomposed into the mode-wise maximization problems represented in the following lemma.

Lemma 1. *An estimate of the parameters* $\mathbf{V}^{(k)}$ *for* $k = 1, 2, \ldots, M$ *in the maximization problem 3.5 can be obtained by solving the following maximization problem for each mode:*

$$\max_{\substack{\mathbf{V}^{(k)} \\ k=1,2,\ldots,M}} \sum_{g=1}^{G} \prod_{k=1}^{M} \operatorname{tr}\left\{\mathbf{V}^{(k)\top}\mathbf{S}_{(g)}^{(k)}\mathbf{V}^{(k)}\mathbf{V}^{(k)\top}\mathbf{S}_{(g)}^{(k)}\mathbf{V}^{(k)}\right\}, \quad \text{s.t.} \quad \mathbf{V}^{(k)\top}\mathbf{V}^{(k)} = \mathbf{I}_{R_k}. \tag{3.6}$$

However, we cannot simultaneously solve this problem for $\mathbf{V}^{(k)}, k = 1, 2, \ldots, M$. Thus, by summarizing the terms unrelated to $\mathbf{V}^{(k)}$ in maximization problem 3.6, we can obtain the maximization problem for kth mode,

$$\max_{\mathbf{V}^{(k)}} f_k(\mathbf{V}^{(k)}) = \max_{\mathbf{V}^{(k)}} \operatorname{tr}\left\{\mathbf{V}^{(k)\top}\mathbf{M}(\mathbf{V}^{(k)})\mathbf{V}^{(k)}\right\}, \quad \text{s.t.} \quad \mathbf{V}^{(k)\top}\mathbf{V}^{(k)} = \mathbf{I}_{R_k}, \tag{3.7}$$

where $\mathbf{M}(\mathbf{V}^{(k)}) = \sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{V}^{(k)}\mathbf{V}^{(k)\top}\mathbf{S}_{(g)}^{(k)}$, in which $w_{(g)}^{(-k)}$ is given by

$$w_{(g)}^{(-k)} = \prod_{j \neq k} \operatorname{tr}\left\{\mathbf{V}^{(j)\top}\mathbf{S}_{(g)}^{(j)}\mathbf{V}^{(j)}\mathbf{V}^{(j)\top}\mathbf{S}_{(g)}^{(j)}\mathbf{V}^{(j)}\right\}. \tag{3.8}$$

Although an estimate of $\mathbf{V}^{(k)}$ can be obtained by solving maximization problem 3.7, this problem is nonconvex, since $\mathbf{V}^{(k)}$ is assumed to be an orthogonal matrix. Thus, the maximization problem has several local maxima. However, by choosing the initial values of parameters in the estimation near the global optimal solution, we can obtain the global optimal solution. In section 4, we develop not only an estimation algorithm but also an initialization method for choosing the initial values of the parameters near the global optimal solution. The initialization method helps guarantee the convergence of our algorithm to the mode-wise global optimal solution.

4 Estimation

Our estimation algorithm consists of two steps: initializing the parameters and iteratively updating the parameters. The initialization step gives us the initial values of the parameters near the global optimal solution for each mode. Next, by iteratively updating the parameters, we can monotonically increase the value of the objective function 3.7 until convergence.

4.1 Initialization. The first step is to initialize the parameters $\mathbf{V}^{(k)}$ for each mode. We define an objective function $f_k'(\mathbf{V}^{(k)}) = \mathrm{tr}\left\{\mathbf{V}^{(k)\top}\mathbf{M}\left(\mathbf{I}^{(k)}\right)\mathbf{V}^{(k)}\right\}$ for $k = 1, \ldots, M$, where $\mathbf{M}\left(\mathbf{I}^{(k)}\right) = \sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{S}_{(g)}^{(k)}$. Next, we adopt a maximizer of $f_k'(\mathbf{V}^{(k)})$ as initial values of the parameters $\mathbf{V}^{(k)}$. To obtain the maximizer, we need an initial value of $\boldsymbol{w}^{(k)} = \left[w_{(1)}^{(-k)}, w_{(2)}^{(-k)}, \ldots, w_{(G)}^{(-k)}\right]$. The initial value for $\boldsymbol{w}^{(k)}$ is obtained by solving the quadratic programming problem,

$$\min_{\boldsymbol{w}^{(k)}} \boldsymbol{w}^{(k)\top}\boldsymbol{\lambda}_0^{(k)}\boldsymbol{\lambda}_0^{(k)\top}\boldsymbol{w}^{(k)}, \quad \text{s.t.} \quad \boldsymbol{w}^{(k)} > \mathbf{0},\ \boldsymbol{w}^{(k)\top}\boldsymbol{\lambda}_1^{(k)}\boldsymbol{\lambda}_1^{(k)\top}\boldsymbol{w}^{(k)} = 1, \tag{4.1}$$

where

$$\boldsymbol{\lambda}_0^{(k)} = \left[\sum_{i=R_k+1}^{P_k}\lambda_{(1)i}^{(k)}, \sum_{i=R_k+1}^{P_k}\lambda_{(2)i}^{(k)}, \ldots, \sum_{i=R_k+1}^{P_k}\lambda_{(G)i}^{(k)}\right]^\top,$$

$$\boldsymbol{\lambda}_1^{(k)} = \left[\sum_{i=1}^{P_k}\lambda_{(1)i}^{(k)}, \sum_{i=1}^{P_k}\lambda_{(2)i}^{(k)}, \ldots, \sum_{i=1}^{P_k}\lambda_{(G)i}^{(k)}\right]^\top, \tag{4.2}$$

in which $\lambda_{(g)i}^{(j)}$ is the ith largest eigenvalue of $\mathbf{S}_{(g)}^{(j)}\mathbf{S}_{(g)}^{(j)}$.

Using the initial value of $\boldsymbol{w}^{(k)}$, we can obtain the initial value of the parameter $\mathbf{V}_0^{(k)}$ by maximizing $f_k'(\mathbf{V}^{(k)})$ for each mode. The maximizer consists of R_k eigenvectors, corresponding to the R_k largest eigenvalues, obtained by eigenvalue decomposition of $\mathbf{M}\left(\mathbf{I}^{(k)}\right)$. The theoretical justification for this initialization is discussed in section 5.

4.2 Iterative Update of Parameters. The second step is to update parameters $\mathbf{V}^{(k)}$ for each mode. We update parameters such that the objective function $f_k(\mathbf{V}^{(k)})$ is maximized. Let $\mathbf{V}_s^{(k)}$ be the value of $\mathbf{V}^{(k)}$ at step s. Then we solve the surrogate maximization problem,

$$\max_{\mathbf{V}_{s+1}^{(k)}} \mathrm{tr}\left\{\mathbf{V}_{s+1}^{(k)\ \top}\mathbf{M}(\mathbf{V}_s^{(k)})\mathbf{V}_{s+1}^{(k)}\right\}, \quad \text{s.t.} \quad \mathbf{V}_{s+1}^{(k)\ \top}\mathbf{V}_{s+1}^{(k)} = \mathbf{I}_{R_k}. \tag{4.3}$$

The solution of equation 4.3 consists of R_k eigenvectors, corresponding to the R_k largest eigenvalues, obtained by eigenvalue decomposition of $\mathbf{M}(\mathbf{V}_s^{(k)})$. By iteratively updating the parameters, the objective function $f_k(\mathbf{V}^{(k)})$ is monotonically increased, which allows it to be maximized. The monotonically increasing property is discussed in section 5.

Our estimation procedure comprises the above estimation steps. The procedure is summarized as algorithm 1.

Algorithm 1: Iteratively Updating Algorithm via Eigenvalue Decomposition.

```
Input: $M$th order tensor data set $\left\{\mathcal{X}_{(g)} \in \mathbb{R}^{P_1 \times P_2 \times \cdots \times P_M \times N_g}, g = 1, 2, \ldots, G\right\}$.
Calculate covariance matrix for tensors: $\mathbf{S}^{*}_{(g)}$ via equations 3.1 and 3.2
Step 1 Initialization:
$\boldsymbol{w}^{(k)} \leftarrow$ the solution of quadratic programming problem equations 4.1, $k = 1, 2, \ldots, M$.
$\mathbf{V}^{(k)}_{0} \leftarrow R_k$ eigenvectors obtained by the eigenvalue decomposition of $\mathbf{M}\left(\mathbf{I}^{(k)}\right)$, $k = 1, 2, \ldots, M$.
$\mathbf{\Lambda}^{(k)}_{(g)} \leftarrow \mathbf{V}^{(k)\top} \mathbf{S}^{(k)}_{(g)} \mathbf{V}^{(k)}, \quad k = 1, 2, \ldots, M; g = 1, 2, \ldots, G$.
Step 2 Updating parameters:
for $s = 1, 2, \ldots$ do
    Update $\mathbf{V}^{(k)}$: $\mathbf{V}^{(k)}_{s+1} \leftarrow R_k$ eigenvectors obtained by eigenvalue decomposition of $\mathbf{M}\left(\mathbf{V}^{(k)}_{s}\right), \quad k = 1, 2, \ldots, M$.
    Update $\mathbf{\Lambda}^{(k)}_{(g)}$: $\mathbf{\Lambda}^{(k)}_{(g)} \leftarrow {\mathbf{V}^{(k)}_{s+1}}^{\top} \mathbf{S}^{(k)}_{(g)} \mathbf{V}^{(k)}_{s+1}, \quad k = 1, 2, \ldots, M; g = 1, 2, \ldots, G$.
return $\mathbf{V}^{(k)} \in \mathbb{R}^{P_k \times R_k}, \mathbf{\Lambda}^{(k)}_{(g)} \in \mathbb{S}^{R_k}_{+}, \quad k = 1, 2, \ldots, M; g = 1, 2, \ldots, G$.
```

5 Theory

This section presents the theoretical and computational analyses for algorithm 1. Theoretical analyses consist of two steps. First, we prove that the initial values of parameters obtained in section 4.1 are relatively close to the global optimal solution. If the initial values are close to the global maximum, then we can obtain the global optimal solution even if the maximization problem is nonconvex. Second, we prove that the iterative updates of the parameters in section 4.2 monotonically increase the value of objective function 3.7 by solving the surrogate problem 4.3. From the monotonically increasing property, the estimated parameters always converge at a stationary point. The combination of these two results enables us to obtain the mode-wise global optimal solution. In the computational analysis, we calculate computational cost for MCCA and then compare the cost with conventional methods. By comparing the costs, we investigate the computational efficacy of MCCA.

5.1 Analysis of Upper and Lower Bounds. This section aims to provide the upper and lower bounds of the maximization problem 3.7. From the bounds, we find that the initial values in section 4.1 are relatively close to the global optimal solution. Before providing the bounds, we define a contraction ratio.

Definition 1. *Let $f_k'^{\max}$ be the global maximum of $f_k'(\mathbf{V}^{(k)})$ and $M^{(k)} = \mathrm{tr}\left\{\mathbf{M}\left(\mathbf{I}^{(k)}\right)\right\}$. Then a contraction ratio of data for kth mode is defined by*

$$\alpha^{(k)} = \frac{f_k'^{\max}}{M^{(k)}} = \frac{\mathrm{tr}\left\{\mathbf{V}_0^{(k)\top}\mathbf{M}\left(\mathbf{I}^{(k)}\right)\mathbf{V}_0^{(k)}\right\}}{\mathrm{tr}\left\{\mathbf{M}\left(\mathbf{I}^{(k)}\right)\right\}}. \tag{5.1}$$

Note that a contraction ratio $\alpha^{(k)}$ satisfies $0 \leq \alpha^{(k)} \leq 1$ and $\alpha^{(k)} = 1$ if and only if $R_k = P_k$.

Using $f_k'^{\max}$ and the contraction ratio $\alpha^{(k)}$, we have the following theorem that reveals the upper and lower bounds of the global maximum in problem 3.7.

Theorem 1. *Let $f_k^{\max}$ be the global maximum of $f_k(\mathbf{V}^{(k)})$. Then*

$$\alpha^{(k)} f_k'^{\max} \leq f_k^{\max} \leq f_k'^{\max}, \tag{5.2}$$

where $\alpha^{(k)}$ is the contraction ratio defined in equation 5.1 and $f_k'^{\max}$ is the global maximum of $f_k'(\mathbf{V}^{(k)})$.

This theorem indicates that $f_k'^{\max} \to f_k^{\max}$ when $\alpha^{(k)} \to 1$. Thus, it is important to obtain an $\alpha^{(k)}$ that is as close as possible to one. Since $\alpha^{(k)}$ depends on $\mathbf{V}_0^{(k)}$ and $\boldsymbol{w}^{(k)}$, $\mathbf{V}_0^{(k)}$ depends on $\boldsymbol{w}^{(k)}$. From this dependency, if we could set the initial value of $\boldsymbol{w}^{(k)}$ such that $\alpha^{(k)}$ is as large as possible, then we could obtain an initial value of $\mathbf{V}_0^{(k)}$ that attains a value near $f_k^{\max}$. The following theorem shows that we can compute the initial value of $\boldsymbol{w}^{(k)}$ such that $\alpha^{(k)}$ is maximized.

Theorem 2. *Let $\boldsymbol{\lambda}_0^{(k)}$ and $\boldsymbol{\lambda}_1^{(k)}$ be the vectors consisting of eigenvalues defined in equation 4.2. For $\boldsymbol{w}^{(k)} = \left[w_{(1)}^{(-k)}, w_{(2)}^{(-k)}, \ldots, w_{(G)}^{(-k)}\right]$ $(k = 1, 2, \ldots, M)$, suppose that the estimate $\hat{\boldsymbol{w}}^{(k)}$ is obtained by solving equation 4.1 for $k = 1, 2, \ldots, M$. Then $\hat{\boldsymbol{w}}^{(k)}$ maximizes $\alpha^{(k)}$.*

In fact, $\alpha^{(k)}$ is very close to one with the initial values given in theorem 2 even if R_k is small. This resembles the cumulative contribution ratio in PCA.

5.2 Convergence Analysis. We next verify that our proposed procedure for iteratively updating parameters maximizes the optimization problem 3.7. In algorithm 1, the parameter $\mathbf{V}_{s+1}^{(k)}$ can be obtained by solving the surrogate maximization problem 4.3. Theorem 3 shows that we can monotonically increase the value of the function $f_k(\mathbf{V}^{(k)})$ in equation 3.7 by algorithm 1.

Theorem 3. *Let* $\mathbf{V}_{s+1}^{(k)}$ *be* R_k *eigenvectors, corresponding to the* R_k *largest eigenvalues, obtained by eigenvalue decomposition of* $\mathbf{M}(\mathbf{V}_s^{(k)})$*. Then*

$$f_k(\mathbf{V}_s^{(k)}) \leq f_k(\mathbf{V}_{s+1}^{(k)}). \tag{5.3}$$

From theorem 1, we obtain initial values of the parameters that are near the global optimal solution. By combining theorems 1 and 3, the solution from algorithm 1 can be characterized by the following corollary.

Corollary 1. *Consider the maximization problem 3.7. Suppose that the initial value of the parameter is obtained by* $\mathbf{V}_0^{(k)} = \underset{\mathbf{V}^{(k)}}{\arg\max}\, \widetilde{f_k}'(\mathbf{V}^{(k)})$*, and the parameter* $\mathbf{V}_s^{(k)}$ *is repeatedly updated by algorithm 1. Then the mode-wise global maximum for the maximization problem 3.7 is achieved when all the contraction ratios* $\alpha^{(k)}$ *for* $k = 1, 2, \ldots, M$ *go to one.*

Algorithm 1 does not guarantee the global solution due to the fundamental problem of nonconvexity, but it is enough for pragmatic purposes. We investigate the issue of convergence to global solution through numerical studies in section 6.3.

5.3 Computational Analysis.

First, we analyze the computational cost. To simplify the analysis, we assume $P = \underset{j}{\arg\max}\, P_j$ for $j = 1, 2, \ldots, M$. This implies that P is the upper bound of R_j for all j. We then calculate the upper bound of the computational complexity.

The expensive computations of the each iteration in algorithm 1 consist of three parts: the formulation of $\mathbf{M}(\mathbf{V}_s^{(k)})$, the eigenvalue decomposition of $\mathbf{M}(\mathbf{V}_s^{(k)})$, and updating latent covariance matrices $\mathbf{\Lambda}_g^{(k)}$. These steps are $O(GM^2P^3)$, $O(P^3)$, and $O(GMP^3)$, respectively. The total computational complexity per iteration is then $O(GM^2P^3)$.

Next, we analyze the memory requirement of algorithm 1. MCCA represents the original tensor data with fewer parameters by projecting the data onto a lower-dimensional space. This requires the $P_k \times R_k$ projection matrices $\mathbf{V}^{(k)}$ for $k = 1, 2, \ldots, M$. MCCA projects the data with size of $N\left(\prod_{k=1}^{M} P_k\right)$ to $N\left(\prod_{k=1}^{M} R_k\right)$, where $N = \sum_{g=1}^{G} N_g$. Thus, the required size for the parameters is $\sum_{k=1}^{M} P_k R_k + N\left(\prod_{k=1}^{M} R_k\right)$. MPCA requires the same amount of

Table 1: Comparisons of the Computational Complexity and the Memory Requirement.

Method	Computational Complexity	Memory Reqirement
PCA	$O(P^{3M})$	$R\left(\prod_{k=1}^{M} P_k\right) + NR$
CCA	$O(GP^{3M})$	$R\left(\prod_{k=1}^{M} P_k\right) + NR$
MPCA	$O(NMP^{M+1})$	$\sum_{k=1}^{M} P_k R_k + N\left(\prod_{k=1}^{M} R_k\right)$
MCCA	$O(GM^2P^3)$	$\sum_{k=1}^{M} P_k R_k + N\left(\prod_{k=1}^{M} R_k\right)$

memory as MCCA. Meanwhile, CCA and PCA need a projection matrix, which is size $R\left(\prod_{k=1}^{M} P_k\right)$. The required size for the parameters is then $R\left(\prod_{k=1}^{M} P_k\right) + NR$.

To compare the computational cost clearly, the upper bounds of computational complexity and the memory requirement are summarized in Table 1. Table 1 shows that the computational complexity of MCCA is superior to that of the other algorithms and the complexity of MCCA is not limited by sample size. In contrast, the MPCA algorithm is affected by the sample size (Lu, Plataniotis, & Venetsanopoulos, 2008). Additionally, MCCA and MPCA require a large amount of memory when the number of modes in a data set is large, but their memory requirements are much smaller than those of PCA and CCA.

6 Experiment

To demonstrate the efficacy of MCCA, we applied MCCA, PCA, CCA, and MPCA to image compression tasks.

6.1 Experimental Setting.

For the experiments, we prepared the following three image data sets:

MNIST data set consists of data of handwritten digits $0, 1, \ldots, 9$ at image sizes of 28 × 28 pixels. The data set includes a training data set of 60,000 images and a test data set of 10,000 images. We used the first 10 training images of the data set for each group. The MNIST data set (Lecun, Bottou, Bengio, & Haffner, 1998) is available at http://yann.lecun.com/exdb/mnist/.

AT&T (ORL) face data set contains gray-scale facial images of 40 people. The data set has 10 images sized 92 × 112 pixels for each person. We used images resized by a factor of 0.5 to improve the efficiency of the

Table 2: Summary of the Data Sets.

Data Set	Group Size	Sample Size (/Group)	Number of Dimensions	Number of Groups
MNIST	Small	10	$28 \times 28 = 784$	10
AT&T(ORL)	Small	10	$46 \times 56 = 2576$	10
	Medium			20
	Large			40
Cropped AR	Small	14	$30 \times 41 \times 3 = 7380$	10
	Medium			25
	Large			50

experiment. The AT&T face data set is available at https://git-disl.github.io/GTDLBench/datasets/att_face_dataset/. All the credits of this data set go to AT&T Laboratories Cambridge.

Cropped AR database has color facial images of 100 people. These images are cropped around the face. The size of images is $120 \times 165 \times 3$ pixels. The data set contains 26 images in each group, 12 of which are images of people wearing sunglasses or scarves. We used the cropped facial images of 50 males who were not wearing sunglasses or scarves. Due to memory limitations, we resized these images by a factor of 0.25. The AR database (Martinez & Benavente, 1998; Martinez & Kak, 2001) is available at http://www2.ece.ohio-state.edu/∼aleix/ARdatabase.html.

The data set characteristics are summarized in Table 2.

To compress these images, we performed dimensionality reductions by MCCA, PCA, CCA, and MPCA, as follows. We vectorized the tensor data set before performing PCA and CCA. In MCCA, the images were compressed and reconstructed according to the following steps:

1. Prepare the multiple image data sets $\mathcal{X}_{(g)} \in \mathbb{R}^{P_1 \times P_2 \times \cdots \times P_M \times N_g}$ for $g = 1, 2, \ldots, G$.
2. Compute the covariance matrix of $\mathcal{X}_{(g)}$ for $g = 1, 2, \ldots, G$.
3. From these covariance matrices, compute the linear transformation matrices $\mathbf{V}_i \in \mathbb{R}^{P_i \times R_i}$ for $i = 1, 2, \ldots, M$ for mapping to the $(R_1, R_2, \ldots, R_M)$-dimensional latent space.
4. Map the ith sample $\mathcal{X}_{(g)i}$ to $\mathcal{X}_{(g)i} \times_1 \mathbf{V}_1 \times_2 \mathbf{V}_2 \cdots \times_M \mathbf{V}_M \in \mathbb{R}^{R_1 \times R_2 \times \cdots \times R_M}$, where the operator $\times_i$ is the i-mode product of tensor (Kolda & Bader, 2009).
5. Reconstruct ith sample $\tilde{\mathcal{X}}_{(g)i} = \mathcal{X}_{(g)i} \times_1 \mathbf{V}_1 \mathbf{V}_1^\top \times_2 \mathbf{V}_2 \mathbf{V}_2^\top \cdots \times_M \mathbf{V}_M \mathbf{V}_M^\top$.

Meanwhile, PCA and MPCA each require a single data set. Thus, we aggregated the data sets as $\mathcal{X} = [\mathcal{X}_{(1)}, \mathcal{X}_{(2)}, \ldots, \mathcal{X}_{(G)}] \in \mathbb{R}^{P_1 \times P_2 \times \cdots \times P_M \times \sum_{g=1}^{G} Ng}$ and performed PCA and MPCA for data set $\mathcal{X}$.

6.2 Performance Assessment. For MCCA and MPCA, the reduced dimensions R_1 and R_2 were chosen as the same number, and then we fixed R_3 as two. All computations were performed by the software `R` (version 3.6) (R Core Team, 2019). In the initialization of MCCA, solving the quadratic programming problem was carried out using the function `ipop` in the package `kernlab`. MPCA was implemented as the function `mpca` in the package `rTensor`. (The implementations of MCCA, PCA, and CCA are available at https://github.com/yoshikawa-kohei/MCCA.)

To assess their performances, we calculated the reconstruction error rate (RER) under the same compression ratio (CR). RER is defined by

$$\mathrm{RER} = \frac{\|\mathcal{X} - \widetilde{\mathcal{X}}\|_F^2}{\|\mathcal{X}\|_F^2}, \tag{6.1}$$

where $\widetilde{\mathcal{X}} = [\widetilde{\mathcal{X}}_{(1)}, \widetilde{\mathcal{X}}_{(2)}, \ldots, \widetilde{\mathcal{X}}_{(G)}]$ is the aggregated data set of reconstructed tensors $\widetilde{\mathcal{X}}_{(g)} = [\widetilde{\mathcal{X}}_{(g)1}, \widetilde{\mathcal{X}}_{(g)2}, \ldots, \widetilde{\mathcal{X}}_{(g)N_g}]$ for $g = 1, 2, \ldots, G$ and $\|\mathcal{X}\|_F$ is the norm of a tensor $\mathcal{X} \in \mathbb{R}^{P_1 \times P_2 \times \cdots \times P_M}$ computed by

$$\|\mathcal{X}\|_F = \sqrt{\sum_{p_1=1}^{P_1} \sum_{p_2=1}^{P_2} \cdots \sum_{p_M=1}^{P_M} x_{p_1, p_2, \ldots, p_M}^2}, \tag{6.2}$$

in which $x_{p_1, p_2, \ldots, p_M}$ is an element $(p_1, p_2, \ldots, p_M)$ of $\mathcal{X}$. In addition, we defined CR as

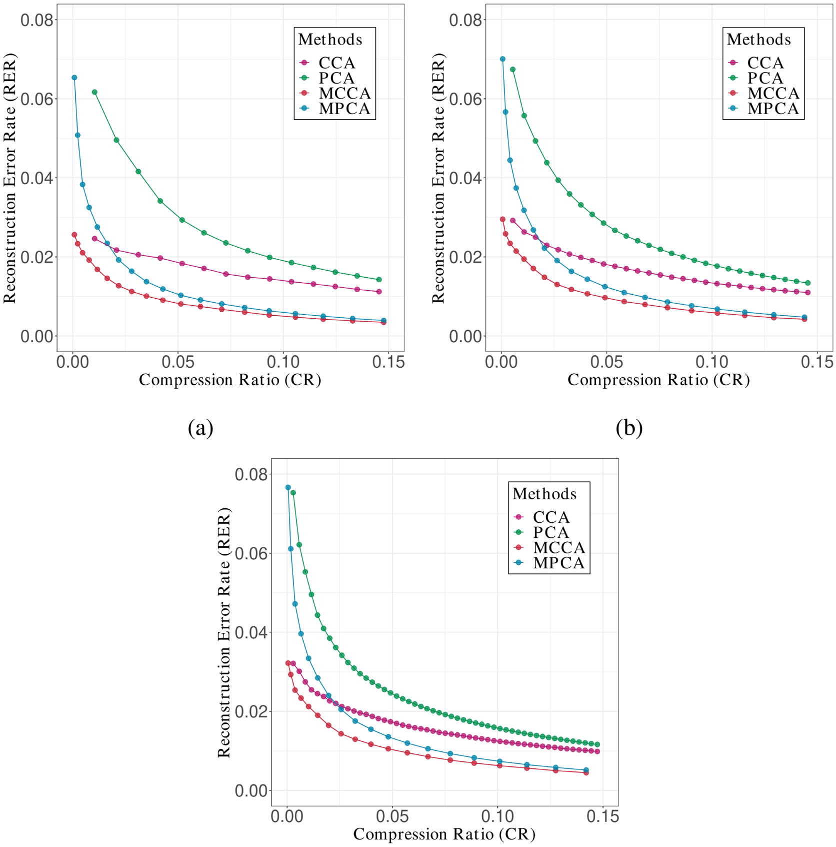
$$\mathrm{CR} = \frac{\{\text{The number of required parameters}\}}{N \cdot \prod_{k=1}^{M} P_k}. \tag{6.3}$$

The number of required parameters for MCCA and MPCA is $\sum_{k=1}^{M} P_k R_k + N\left(\prod_{k=1}^{M} R_k\right)$, whereas that for CCA and PCA is $R\left(\prod_{k=1}^{M} P_k\right) + NR$.

Figure 1 plots the RER obtained by estimating various reduced dimensions for the AT&T(ORL) data set with group sizes of small, medium, and large. As the figures for the results of the other data sets were similar to Figure 1, we show them in the supplementary materials S1.

From Figure 1, we observe that the RER material MCCA is the smallest for any value of CR. This indicates that MCCA performs better than the other methods. In addition, note that CCA performs better than MPCA only for fairly small values of CR, even though it is a method for vector data, whereas MPCA performs better for larger values of CR. This implies the limitations of CCA for vector data.

Next we consider group size by comparing panels a, b, and c in Figure 1. The value of CR at the intersection of CCA and MPCA increases with



(a)

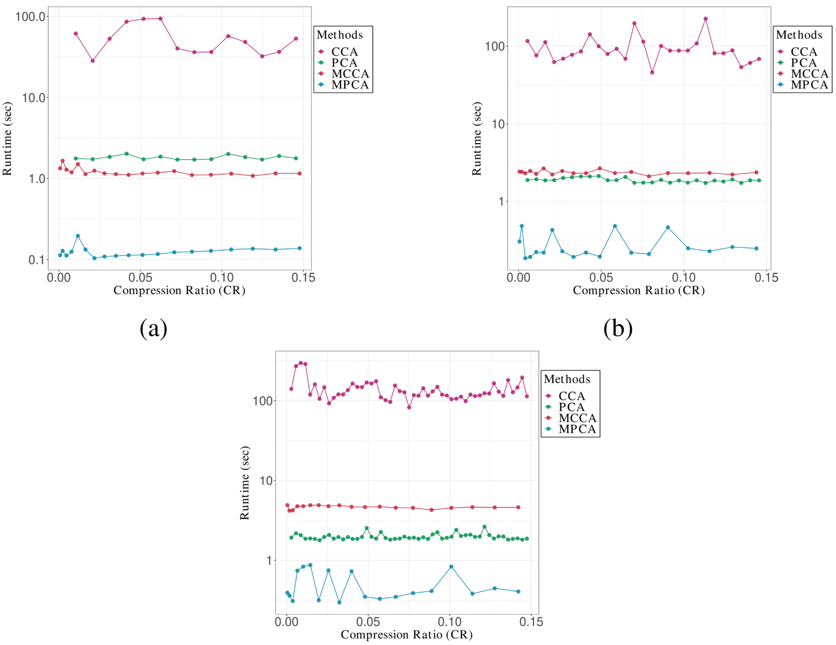
(b)

(c)

Figure 1: Plots of RER versus CR for the AT&T(ORL) data set of various group sizes: (a) small, (b) medium, and (c) large.

increasing the group size. This indicates that MPCA has more trouble extracting an appropriate latent space as the group size increases. Since MPCA does not consider the group structure, it is not possible to properly estimate the covariance structure when the group size is large.

Figure 2 shows the comparison of runtime for the AT&T(ORL) data set with group sizes of small, medium, and large. Although Table 1 gives the superiority of the computational complexity for MCCA, Figure 2 shows that MCCA is slower than MPCA for any size of data set. This probably arises from the difference of implementation of MCCA and MPCA: MCCA is implemented by our hand-built source code, while MPCA is done by the package `rTensor`. But when we compare MCCA with CCA, MCCA is



(a)

(b)

(c)

Figure 2: Plots of runtime versus CR for the AT&T(ORL) data set of various group sizes: (a) small, (b) medium, and (c) large.

superior to CCA in terms of both computational complexity and the runtime comparisons.

Figure 3 plots the reconstructed images for the AT&T(ORL) data set with group sizes of the medium. This figure can be obtained by performing four methodologies when we set $R_1 = R_2 = 5$ and $R = 2$. By setting the number of the ranks in this way, we can compare the images with almost the same CR, PCA, CCA, and MPCA can recover the average structure of face images, but they cannot deal with changes in the angle of the face. MCCA can also recover the detailed differences in each image.

6.3 Behavior of Contraction Ratio. We examined the behavior of contraction ratio $\alpha^{(k)}$. We performed MCCA on the AT&T(ORL) data set with the medium group size and computed $\alpha^{(1)}$ and $\alpha^{(2)}$ with the various pairs of reduced dimensions $(R_1, R_2) \in \{1, 2, \ldots, 25\} \times \{1, 2, \ldots, 25\}$.

Figure 4 shows the values of $\alpha^{(1)}$ and $\alpha^{(2)}$ for all pairs of R_1 and R_2. As shown, $\alpha^{(1)}$ and $\alpha^{(2)}$ were invariant to variations in R_2 and R_1, respectively. Therefore, to facilitate visualization of changes in $\alpha^{(k)}$, we draw Figure 5,

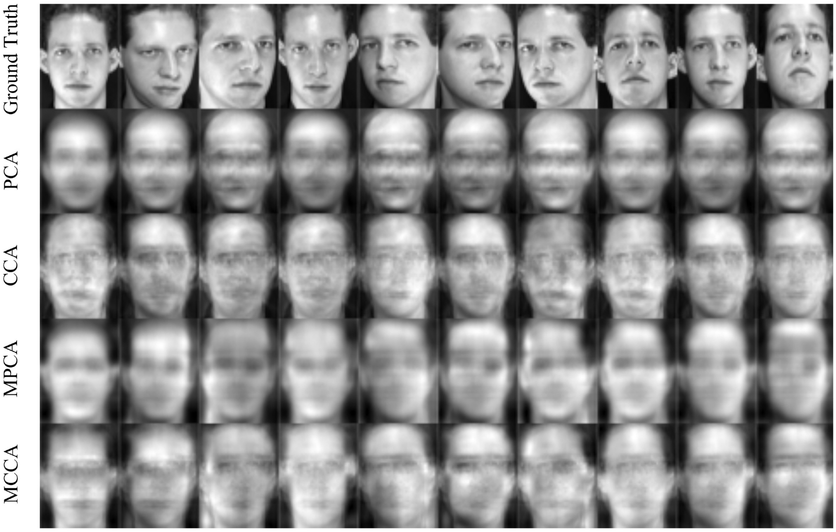


Figure 3: The reconstructed images for the AT&T(ORL) data set with the medium group sizes under almost the same CR. Image source: AT&T Laboratories Cambridge.

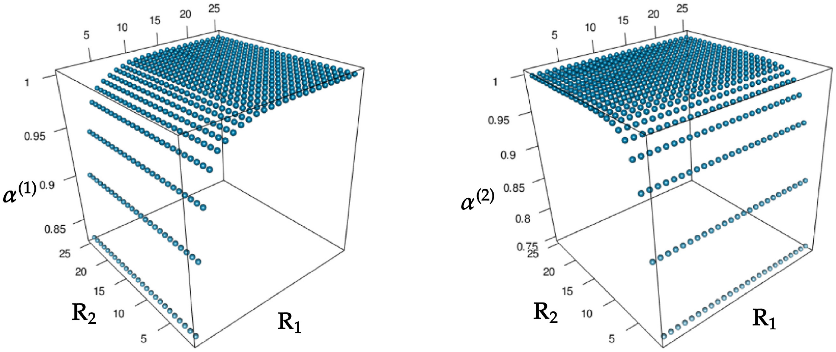


Figure 4: $\alpha^{(1)}$ and $\alpha^{(2)}$ versus pairs of reduced dimensions (R_1, R_2).

which represents $\alpha^{(1)}$ and $\alpha^{(2)}$ for, respectively, $R_2 = 1$ and $R_1 = 1$. From these, we observe that when both R_1 and R_2 are greater than eight, both $\alpha^{(1)}$ and $\alpha^{(2)}$ are close to one.

6.4 Efficacy of Solving the Quadratic Programming Problem. We investigated the usefulness of determining the initial value of $\boldsymbol{w}^{(k)}$ by solving the quadratic programming problem 4.1. We applied MCCA to the

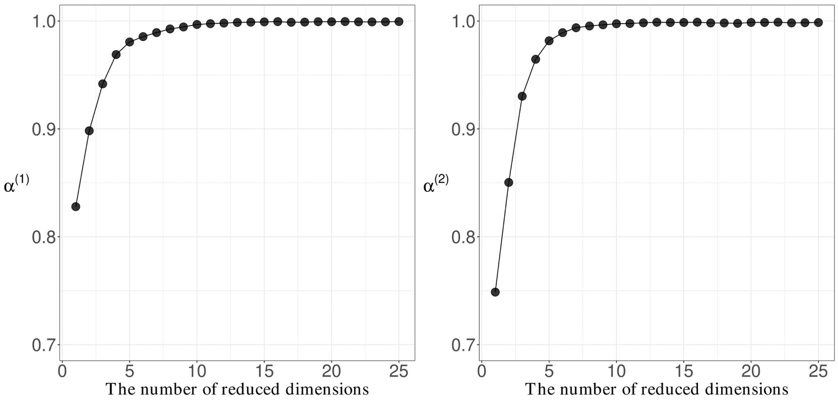


Figure 5: $\alpha^{(1)}$ and $\alpha^{(2)}$ versus R_1 and R_2, respectively.

AT&T(ORL) data set with the small, medium, and large number of groups. In addition, we used the smaller group size of three. For determining the initial value of $\boldsymbol{w}^{(k)}$, we consider three methods: solving the quadratic programming problem 4.1 (MCCA:QP); setting all values of $\boldsymbol{w}^{(k)}$ to one (MCCA:FIX); and setting the values by random sampling according to the uniform distribution $U(0, 1)$ (MCCA:RANDOM). We computed the $\alpha^{(k)}$ with the reduced dimensions $R_1 = R_2$ ($\in \{1, 2, \ldots, 10\}$) for each of these methods.

To evaluate the performance of these methods, we compared the values of $\alpha^{(k)}$ and the number of iterations in the estimation. The number of iterations in the estimation is the number of repetitions of lines 7 to 9 in algorithm 1. For MCCA(RANDOM), we performed 50 trials and calculated averages of each of these indices.

Figure 6 shows the comparisons of $\alpha^{(1)}$ and $\alpha^{(2)}$ when the initialization was performed by MCCA:QP, MCCA:FIX, and MCCA:RANDOM for the AT&T(ORL) data set with a group size of three. It was confirmed that MCCA:QP provides the largest values of $\alpha^{(1)}$ and $\alpha^{(2)}$. Figure 7 shows the number of iterations. MCCA:QP gives the smallest number of iterations for almost all values of the reduced dimensions. This result indicates that MCCA:QP converges to a solution faster than the other initialization methods. However, when the reduced dimension is greater than or equal to eight, the other methods are competitive with MCCA:QP. A lack of difference in the number of iterations could result from the closeness of the initial values and the global optimal solution. Note that when the R_1 and R_2 are greater than or equal to eight, $\alpha^{(1)}$ and $\alpha^{(2)}$ are sufficiently close to one, based on Figure 6. This indicates that the initial values are close to the global optimal solution obtained from theorem 1. Hence, the result shows almost the same number of iterations for the three methods.

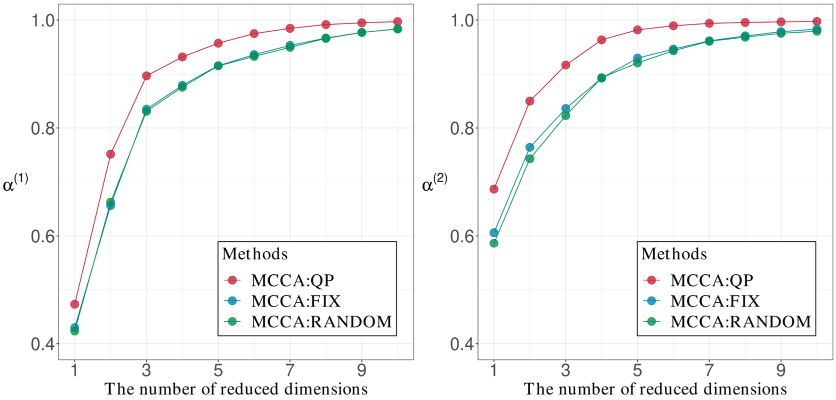


Figure 6: Comparisons of $\alpha^{(1)}$ and $\alpha^{(2)}$ computed by using the initial values obtained from the initializations MCCA:QP, MCCA:FIX, and MCCA:RANDOM with the AT&T(ORL) data set for a group size of three.

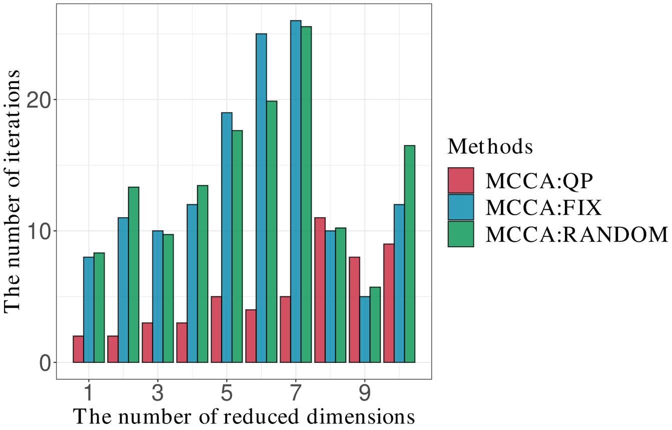


Figure 7: Comparison of the number of iterations when the initialization was performed by MCCA:QP, MCCA:FIX, and MCCA:RANDOM with the AT&T(ORL) data set for a group size of three.

Figures 8 and 9 show comparisons for the AT&T(ORL) data set with the medium group size. Since the figures for the results of other group sizes are similar to Figures 8 and 9, we show them in the supplementary materials S2. Figure 8 shows results similar those in Figure 6, whereas Figure 9 shows competitive performances for all reduced dimensions.

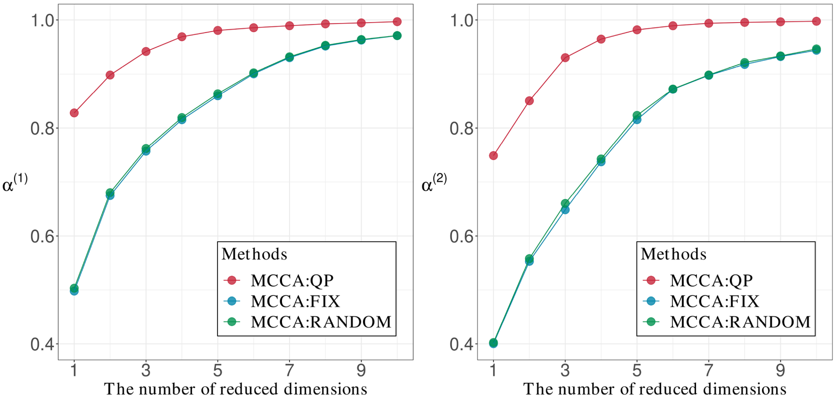


Figure 8: Comparisons of $\alpha^{(1)}$ and $\alpha^{(2)}$ computed using the initial values obtained from the initialization of MCCA:QP, MCCA:FIX, and MCCA:RANDOM with the AT&T(ORL) data set and the medium group size.

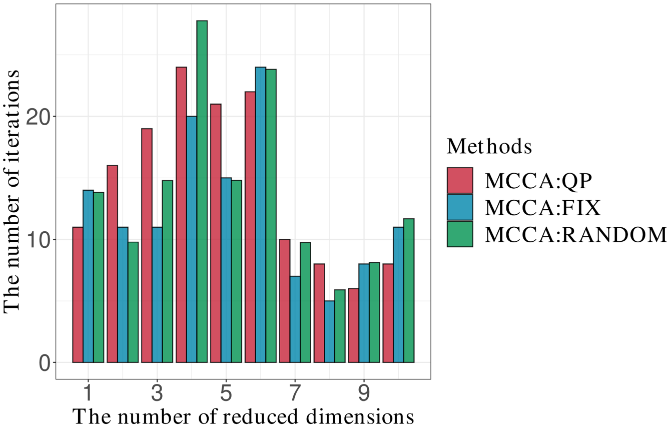


Figure 9: Comparison of the number of iterations when the initialization was perfomed by MCCA:QP, MCCA:FIX, and MCCA:RANDOM with the AT&T(ORL) data set and the medium group size.

7 Conclusion

We have developed the multilinear common components analysis (MCCA) by introducing a covariance structure based on the Kronecker product. To efficiently solve the nonconvex optimization problem for MCCA, we have proposed an iteratively updating algorithm that exhibits some superior theoretical convergence properties. Numerical experiments have shown the usefulness of MCCA.

Specifically, MCCA was shown to be competitive among the initialization methods in terms of the number of iterations. As the number of groups increases, the overall number of samples increases. This may be the reason why all methods required almost the same number of iterations for small, medium, and large groups. Note that in this study, we used the Kronecker product representation to estimate the covariance matrix for tensor data sets. Greenewald, Zhou, and Hero (2019) used the Kronecker sum representation for estimating the covariance matrix, and it would be interesting to extend the MCCA to this and other covariance representations.

Appendix A: Proof of Lemma 1

We provide two basic lemmas about Kronecker products before we prove lemma 1.

Lemma 2. *For matrices* $\mathbf{A}, \mathbf{B}, \mathbf{C}$, *and* $\mathbf{D}$ *such that matrix products* $\mathbf{AC}$ *and* $\mathbf{BD}$ *can be calculated, the following equation holds:*

$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = \mathbf{AC} \otimes \mathbf{BD}.$$

Lemma 3. *For square matrices* $\mathbf{A}$ *and* $\mathbf{B}$, *the following equation holds:*

$$\mathrm{tr}(\mathbf{A} \otimes \mathbf{B}) = \mathrm{tr}(\mathbf{A})\mathrm{tr}(\mathbf{B}).$$

These lemmas are known as the mixed-product property and the spectrum property, respectively. See Harville (1998) for detailed proofs.

Proof of Lemma 1. For the maximization problem 3.5, move the summation over index g out of the $\mathrm{tr}(\cdot)$ and replace $\mathbf{S}^*_{(g)}$ and $\mathbf{V}^*$ with $\mathbf{S}^{(1)}_{(g)} \otimes \mathbf{S}^{(2)}_{(g)} \otimes \cdots \otimes \mathbf{S}^{(M)}_{(g)}$ and $\mathbf{V}^{(1)} \otimes \mathbf{V}^{(2)} \otimes \cdots \otimes \mathbf{V}^{(M)}$, respectively. Then

$$\max_{\substack{\mathbf{V}^{(k)} \\ k=1,2,\ldots,M}} \sum_{g=1}^{G} \mathrm{tr}\left\{ \left(\mathbf{V}^{(1)} \otimes \cdots \otimes \mathbf{V}^{(M)}\right)^\top \left(\mathbf{S}^{(1)}_{(g)} \otimes \cdots \otimes \mathbf{S}^{(M)}_{(g)}\right) \left(\mathbf{V}^{(1)} \otimes \cdots \otimes \mathbf{V}^{(M)}\right) \right.$$
$$\left. \left(\mathbf{V}^{(1)} \otimes \cdots \otimes \mathbf{V}^{(M)}\right)^\top \left(\mathbf{S}^{(1)}_{(g)} \otimes \cdots \otimes \mathbf{S}^{(M)}_{(g)}\right) \left(\mathbf{V}^{(1)} \otimes \cdots \otimes \mathbf{V}^{(M)}\right) \right\},$$
$$\text{s.t.} \quad {\mathbf{V}^{(k)}}^\top \mathbf{V}^{(k)} = \mathbf{I}_{R_k}.$$

By lemmas 2 and 3, we have

$$\begin{aligned}
&\max_{\substack{\mathbf{V}^{(k)\top}\mathbf{V}^{(k)}=\mathbf{I}_{R_k}\\ k=1,2,\ldots,M}} \sum_{g=1}^{G} \mathrm{tr}\left\{\left(\mathbf{V}^{(1)\top}\mathbf{S}_{(g)}^{(1)}\mathbf{V}^{(1)}\mathbf{V}^{(1)\top}\mathbf{S}_{(g)}^{(1)}\mathbf{V}^{(1)}\right)\right.\\
&\qquad\qquad\qquad \left.\cdots\left(\mathbf{V}^{(M)\top}\mathbf{S}_{(g)}^{(M)}\mathbf{V}^{(M)}\mathbf{V}^{(M)\top}\mathbf{S}_{(g)}^{(M)}\mathbf{V}^{(M)}\right)\right\}\\
&= \max_{\substack{\mathbf{V}^{(k)\top}\mathbf{V}^{(k)}=\mathbf{I}_{R_k}\\ k=1,2,\ldots,M}} \sum_{g=1}^{G}\prod_{k=1}^{M} \mathrm{tr}\left\{\mathbf{V}^{(k)\top}\mathbf{S}_{(g)}^{(k)}\mathbf{V}^{(k)}\mathbf{V}^{(k)\top}\mathbf{S}_{(g)}^{(k)}\mathbf{V}^{(k)}\right\}.
\end{aligned}$$

This leads to the maximization problem in lemma 1. □

Appendix B: Proof of Theorem 1

Theorem 1 can be easily shown from the following lemma.

Lemma 4. *Consider the maximization problem*

$$\max_{\mathbf{V}^{(k)}} f_k'(\mathbf{V}^{(k)}) = \max_{\mathbf{V}^{(k)}} \mathrm{tr}\left\{\mathbf{V}^{(k)\top}\left(\sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{S}_{(g)}^{(k)}\right)\mathbf{V}^{(k)}\right\}. \tag{B.1}$$

Let $M^{(k)} = \mathrm{tr}\left\{\sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{S}_{(g)}^{(k)}\right\}$. *Then*

$$\frac{f_k'(\mathbf{V}^{(k)})^2}{M^{(k)}} \le f_k(\mathbf{V}^{(k)}) \le f_k'(\mathbf{V}^{(k)}).$$

Proof of Lemma 4. First, we prove $f_k(\mathbf{V}^{(k)}) \le f_k'(\mathbf{V}^{(k)})$. For any orthogonal matrix $\mathbf{V}^{(k)} \in \mathbb{R}^{P_k \times R_k}$, we can always find an orthogonal matrix $\mathbf{V}_{\perp}^{(k)} \in \mathbb{R}^{P_k \times (P_k - R_k)}$ that satisfies $\mathbf{V}^{(k)\top}\mathbf{V}_{\perp}^{(k)} = \boldsymbol{O}$. Then the equation $\mathbf{V}^{(k)}\mathbf{V}^{(k)\top} + \mathbf{V}_{\perp}^{(k)}\mathbf{V}_{\perp}^{(k)\top} = \mathbf{I}_{P_k}$ holds. By definition,

$$\begin{aligned}
f_k(\mathbf{V}^{(k)}) &= \mathrm{tr}\left\{\mathbf{V}^{(k)\top}\left(\sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{V}^{(k)}\mathbf{V}^{(k)\top}\mathbf{S}_{(g)}^{(k)}\right)\mathbf{V}^{(k)}\right\}\\
&\le \mathrm{tr}\left\{\mathbf{V}^{(k)\top}\left(\sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\left(\mathbf{V}^{(k)}\mathbf{V}^{(k)\top} + \mathbf{V}_{\perp}^{(k)}\mathbf{V}_{\perp}^{(k)\top}\right)\mathbf{S}_{(g)}^{(k)}\right)\mathbf{V}^{(k)}\right\}
\end{aligned}$$

$$\begin{aligned} &= \mathrm{tr}\left\{ \mathbf{V}^{(k)\top} \left(\sum_{g=1}^{G} w_{(g)}^{(-k)} \mathbf{S}_{(g)}^{(k)} \mathbf{S}_{(g)}^{(k)} \right) \mathbf{V}^{(k)} \right\} \\ &= f_k'(\mathbf{V}^{(k)}). \end{aligned}$$

Thus, we have obtained $f_k(\mathbf{V}^{(k)}) \leq f_k'(\mathbf{V}^{(k)})$.

Next, we prove $\frac{f_k'(\mathbf{V}^{(k)})^2}{M^{(k)}} \leq f_k(\mathbf{V}^{(k)})$. We define the following block matrices:

$$\begin{aligned} \mathbf{A} &= \left[\sqrt{w_{(1)}^{(-k)}} \mathbf{S}_{(1)}^{(k)\,\frac{1}{2}} \mathbf{V}^{(k)} \mathbf{V}^{(k)\top} \mathbf{S}_{(1)}^{(k)\,\frac{1}{2}}, \ldots, \sqrt{w_{(G)}^{(-k)}} \mathbf{S}_{(G)}^{(k)\,\frac{1}{2}} \mathbf{V}^{(k)} \mathbf{V}^{(k)\top} \mathbf{S}_{(G)}^{(k)\,\frac{1}{2}} \right], \\ \mathbf{B} &= \left[\sqrt{w_{(1)}^{(-k)}} \mathbf{S}_{(1)}^{(k)}, \ldots, \sqrt{w_{(G)}^{(-k)}} \mathbf{S}_{(G)}^{(k)} \right]. \end{aligned}$$

Note that since $\mathbf{S}_{(g)}^{(k)}$ is a symmetric positive-definite matrix, $\mathbf{S}_{(g)}^{(k)}$ can be decomposed to $\mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \mathbf{S}_{(g)}^{(k)\,\frac{1}{2}}$. We calculate the traces of **AA**, **AB**, and **BB**, respectively:

$$\begin{aligned} \mathrm{tr}\,(\mathbf{AA}) &= \sum_{g=1}^{G} w_{(g)}^{(-k)} \mathrm{tr} \left(\mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \mathbf{V}^{(k)} \mathbf{V}^{(k)\top} \mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \mathbf{V}^{(k)} \mathbf{V}^{(k)\top} \mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \right) \\ &= \sum_{g=1}^{G} w_{(g)}^{(-k)} \mathrm{tr} \left\{ \mathbf{V}^{(k)\top} \mathbf{S}_{(g)}^{(k)} \mathbf{V}^{(k)} \mathbf{V}^{(k)\top} \mathbf{S}_{(g)}^{(k)} \mathbf{V}^{(k)} \right\} \\ &= \mathrm{tr} \left\{ \mathbf{V}^{(k)\top} \left(\sum_{g=1}^{G} w_{(g)}^{(-k)} \mathbf{S}_{(g)}^{(k)} \mathbf{V}^{(k)} \mathbf{V}^{(k)\top} \mathbf{S}_{(g)}^{(k)} \right) \mathbf{V}^{(k)} \right\} \\ &= f_k(\mathbf{V}^{(k)}), \end{aligned}$$

$$\begin{aligned} \mathrm{tr}\,(\mathbf{AB}) &= \sum_{g=1}^{G} w_{(g)}^{(-k)} \mathrm{tr} \left(\mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \mathbf{V}^{(k)} \mathbf{V}^{(k)\top} \mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \mathbf{S}_{(g)}^{(k)} \right) \\ &= \sum_{g=1}^{G} w_{(g)}^{(-k)} \mathrm{tr} \left(\mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \mathbf{V}^{(k)} \mathbf{V}^{(k)\top} \mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \mathbf{S}_{(g)}^{(k)\,\frac{1}{2}} \right) \\ &= \sum_{g=1}^{G} w_{(g)}^{(-k)} \mathrm{tr} \left\{ \mathbf{V}^{(k)\top} \mathbf{S}_{(g)}^{(k)} \mathbf{S}_{(g)}^{(k)} \mathbf{V}^{(k)} \right\} \end{aligned}$$

$$= \mathrm{tr}\left\{\mathbf{V}^{(k)\top}\left(\sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{S}_{(g)}^{(k)}\right)\mathbf{V}^{(k)}\right\}$$

$$= f_k'(\mathbf{V}^{(k)}),$$

$$\mathrm{tr}\,(\mathbf{BB}) = \mathrm{tr}\left(\sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{S}_{(g)}^{(k)}\right) = M^{(k)}.$$

From the Cauchy–Schwarz inequality, we have

$$f_k(\mathbf{V}^{(k)})M^{(k)} = \mathrm{tr}\,(\mathbf{AA})\,\mathrm{tr}\,(\mathbf{BB}) \geq \left\{\mathrm{tr}\,(\mathbf{AB})\right\}^2 = f_k'(\mathbf{V}^{(k)})^2.$$

By dividing both sides of the inequality by $M^{(k)}$, we obtain $\frac{f_k'(\mathbf{V}^{(k)})^2}{M^{(k)}} \leq f_k(\mathbf{V}^{(k)})$. □

Proof of Theorem 1. Let $f_k'^{\max}$ be the global maximum of $f_k'(\mathbf{V}^{(k)})$ and $\mathbf{V}_0^{(k)} = \underset{\mathbf{V}^{(k)}}{\arg\max}\, f_k'(\mathbf{V}^{(k)})$. From lemma 4 and the definition of $\alpha^{(k)}$, we have

$$\alpha^{(k)} f_k'^{\max} = \frac{f_k'(\mathbf{V}_0^{(k)})^2}{M^{(k)}} \leq f_k(\mathbf{V}_0^{(k)}).$$

Let $f_k^{\max}$ be the global maximum of $f_k(\mathbf{V}^{(k)})$. It then holds that $f_k(\mathbf{V}_0^{(k)}) \leq f_k^{\max}$. Thus,

$$\alpha^{(k)} f_k'^{\max} \leq f_k^{\max}.$$

Let $\mathbf{V}_{0^*}^{(k)} = \underset{\mathbf{V}^{(k)}}{\arg\max}\, f_k(\mathbf{V}^{(k)})$. From lemma 4, we have

$$f_k^{\max} = f_k(\mathbf{V}_{0^*}^{(k)}) \leq f_k'(\mathbf{V}_{0^*}^{(k)}).$$

Since $f_k'(\mathbf{V}_{0^*}^{(k)}) \leq f_k'^{\max}$, we have

$$f_k^{\max} \leq f_k'^{\max}.$$

Hence, we have obtained $\alpha^{(k)} f_k'^{\max} \leq f_k^{\max} \leq f_k'^{\max}$. □

Appendix C: Proof of Theorem 2

Proof of Theorem 2. By definition

$$\alpha^{(k)} = \frac{f_k'^{\max}}{M^{(k)}} = \frac{\mathrm{tr}\left\{\mathbf{V}_0^{(k)\top}\left(\sum_{g=1}^{G} w_{(g)}^{(-k)} \mathbf{S}_{(g)}^{(k)} \mathbf{S}_{(g)}^{(k)}\right)\mathbf{V}_0^{(k)}\right\}}{\mathrm{tr}\left\{\sum_{g=1}^{G} w_{(g)}^{(-k)} \mathbf{S}_{(g)}^{(k)} \mathbf{S}_{(g)}^{(k)}\right\}}. \tag{C.1}$$

By using the eigenvalue representation, we can rewrite the numerator of $\alpha^{(k)}$ as

$$f_k'^{\max} = \sum_{g=1}^{G} w_{(g)}^{(-k)} \sum_{i=1}^{R_k} \lambda_{(g)i}^{(k)}.$$

The denominator of $\alpha^{(k)}$ can be represented as the sum of eigenvalues as follows:

$$M^{(k)} = \sum_{g=1}^{G} w_{(g)}^{(-k)} \sum_{i=1}^{P_k} \lambda_{(g)i}^{(k)}.$$

Thus, we can transform $\alpha^{(k)}$ as follows:

$$\alpha^{(k)} = \frac{\sum_{g=1}^{G} w_{(g)}^{(-k)} \sum_{i=1}^{R_k} \lambda_{(g)i}^{(k)}}{\sum_{g=1}^{G} w_{(g)}^{(-k)} \sum_{i=1}^{P_k} \lambda_{(g)i}^{(k)}}.$$

When we set

$$\boldsymbol{\lambda}_0^{(k)} = \left[\sum_{i=R_k+1}^{P_k} \lambda_{(1)i}^{(k)}, \sum_{i=R_k+1}^{P_k} \lambda_{(2)i}^{(k)}, \ldots, \sum_{i=R_k+1}^{P_k} \lambda_{(G)i}^{(k)}\right]^\top,$$

$$\boldsymbol{\lambda}_1^{(k)} = \left[\sum_{i=1}^{P_k} \lambda_{(1)i}^{(k)}, \sum_{i=1}^{P_k} \lambda_{(2)i}^{(k)}, \ldots, \sum_{i=1}^{P_k} \lambda_{(G)i}^{(k)}\right]^\top,$$

$$\boldsymbol{w}^{(k)} = \left[w_{(1)}^{(-k)}, w_{(2)}^{(-k)}, \ldots, w_{(G)}^{(-k)}\right]^\top,$$

we can reformulate $\alpha^{(k)}$ as

$$\alpha^{(k)} = \frac{\left(\boldsymbol{\lambda}_1^{(k)} - \boldsymbol{\lambda}_0^{(k)}\right)^\top \boldsymbol{w}^{(k)}}{\boldsymbol{\lambda}_1^{(k)\top} \boldsymbol{w}^{(k)}}.$$

Thus, we obtain the following maximization problem:

$$\max_{\boldsymbol{w}^{(k)}} \frac{\left(\boldsymbol{\lambda}_1^{(k)} - \boldsymbol{\lambda}_0^{(k)}\right)^\top \boldsymbol{w}^{(k)}}{\boldsymbol{\lambda}_1^{(k)\top} \boldsymbol{w}^{(k)}}, \quad \text{s.t.} \quad \boldsymbol{w}^{(k)} > \mathbf{0}.$$

Note that the constraints can be obtained by the definition of $\boldsymbol{w}^{(k)}$. In addition, this maximization problem can be reformulated as

$$\max_{\boldsymbol{w}^{(k)}} \frac{\left(\boldsymbol{\lambda}_1^{(k)} - \boldsymbol{\lambda}_0^{(k)}\right)^\top \boldsymbol{w}^{(k)}}{\boldsymbol{\lambda}_1^{(k)\top} \boldsymbol{w}^{(k)}} = \max_{\boldsymbol{w}^{(k)}} 1 - \frac{\boldsymbol{\lambda}_0^{(k)\top} \boldsymbol{w}^{(k)}}{\boldsymbol{\lambda}_1^{(k)\top} \boldsymbol{w}^{(k)}} = \min_{\boldsymbol{w}^{(k)}} \frac{\boldsymbol{\lambda}_0^{(k)\top} \boldsymbol{w}^{(k)}}{\boldsymbol{\lambda}_1^{(k)\top} \boldsymbol{w}^{(k)}}.$$

Since $\boldsymbol{\lambda}_0^{(k)\top} \boldsymbol{w}^{(k)} / \boldsymbol{\lambda}_1^{(k)\top} \boldsymbol{w}^{(k)}$ is nonnegative, solving the optimization problem for the squared function of the objective function maintains generality. Thus, we can consider the following minimization problem:

$$\min_{\boldsymbol{w}^{(k)}} \frac{\boldsymbol{w}^{(k)\top} \boldsymbol{\lambda}_0^{(k)} \boldsymbol{\lambda}_0^{(k)\top} \boldsymbol{w}^{(k)}}{\boldsymbol{w}^{(k)\top} \boldsymbol{\lambda}_1^{(k)} \boldsymbol{\lambda}_1^{(k)\top} \boldsymbol{w}^{(k)}}, \quad \text{s.t.} \quad \boldsymbol{w}^{(k)} > \mathbf{0}.$$

Additionally, from the invariance under multiplication of $\boldsymbol{w}^{(k)}$ by a constant, we obtain the following objective function of the quadratic programming problem:

$$\min_{\boldsymbol{w}^{(k)}} \boldsymbol{w}^{(k)\top} \boldsymbol{\lambda}_0^{(k)} \boldsymbol{\lambda}_0^{(k)\top} \boldsymbol{w}^{(k)}, \quad \text{s.t.} \quad \boldsymbol{w}^{(k)} > \mathbf{0}, \; \boldsymbol{w}^{(k)\top} \boldsymbol{\lambda}_1^{(k)} \boldsymbol{\lambda}_1^{(k)\top} \boldsymbol{w}^{(k)} = 1.$$

□

Appendix D: Proof of Theorem 3

Proof of Theorem 3. We define the following block matrices:

$$\mathbf{A}_s = \left[\sqrt{w_{(1)}^{(-k)}} \mathbf{S}_{(1)}^{(k)\,\frac{1}{2}} \mathbf{V}_s^{(k)} \mathbf{V}_s^{(k)\top} \mathbf{S}_{(1)}^{(k)\,\frac{1}{2}}, \ldots, \sqrt{w_{(G)}^{(-k)}} \mathbf{S}_{(G)}^{(k)\,\frac{1}{2}} \mathbf{V}_s^{(k)} \mathbf{V}_s^{(k)\top} \mathbf{S}_{(G)}^{(k)\,\frac{1}{2}} \right].$$

Here, we calculate the traces of $\mathbf{A}_s\mathbf{A}_s$, $\mathbf{A}_s\mathbf{A}_{s+1}$, and $\mathbf{A}_{s+1}\mathbf{A}_{s+1}$. The calculations of $\mathrm{tr}\,(\mathbf{A}_s\mathbf{A}_s)$ and $\mathrm{tr}\,(\mathbf{A}_{s+1}\mathbf{A}_{s+1})$ are the same as that of $\mathrm{tr}\,(\mathbf{A}\mathbf{A})$ by replacing $\mathbf{V}^{(k)}$ with $\mathbf{V}_s^{(k)}$ and $\mathbf{V}^{(k)}$ with $\mathbf{V}_{s+1}^{(k)}$, respectively, in lemma 4. Thus, we obtain

$$\mathrm{tr}\,(\mathbf{A}_s\mathbf{A}_s) = f_k(\mathbf{V}_s^{(k)}),$$

$$\begin{aligned}
\operatorname{tr}\left(\mathbf{A}_s\mathbf{A}_{s+1}\right) &= \sum_{g=1}^{G} w_{(g)}^{(-k)} \operatorname{tr}\left(\mathbf{S}_{(g)}^{(k)\,\frac{1}{2}}\mathbf{V}_s^{(k)}\mathbf{V}_s^{(k)\top}\mathbf{S}_{(g)}^{(k)\,\frac{1}{2}}\mathbf{S}_{(g)}^{(k)\,\frac{1}{2}}\mathbf{V}_{s+1}^{(k)}\mathbf{V}_{s+1}^{(k)}{}^{\top}\mathbf{S}_{(g)}^{(k)\,\frac{1}{2}}\right)\\
&= \sum_{g=1}^{G} w_{(g)}^{(-k)} \operatorname{tr}\left\{\mathbf{V}_{s+1}^{(k)}{}^{\top}\mathbf{S}_{(g)}^{(k)}\mathbf{V}_s^{(k)}\mathbf{V}_s^{(k)\top}\mathbf{S}_{(g)}^{(k)}\mathbf{V}_{s+1}^{(k)}\right\}\\
&= \operatorname{tr}\left\{\mathbf{V}_{s+1}^{(k)}{}^{\top}\left(\sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{V}_s^{(k)}\mathbf{V}_s^{(k)\top}\mathbf{S}_{(g)}^{(k)}\right)\mathbf{V}_{s+1}^{(k)}\right\},
\end{aligned}$$

$$\operatorname{tr}\left(\mathbf{A}_{s+1}\mathbf{A}_{s+1}\right) = f_k(\mathbf{V}_{s+1}^{(k)}).$$

Since $\mathbf{V}_{s+1}^{(k)} = \underset{\mathbf{V}^{(k)}}{\arg\max} \operatorname{tr}\left\{\mathbf{V}^{(k)\top}\left(\sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{V}_s^{(k)}\mathbf{V}_s^{(k)\top}\mathbf{S}_{(g)}^{(k)}\right)\mathbf{V}^{(k)}\right\}$, we have

$$\begin{aligned}
f_k(\mathbf{V}_s^{(k)}) &= \operatorname{tr}\left\{\mathbf{V}_s^{(k)\top}\left(\sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{V}_s^{(k)}\mathbf{V}_s^{(k)\top}\mathbf{S}_{(g)}^{(k)}\right)\mathbf{V}_s^{(k)}\right\}\\
&\leq \operatorname{tr}\left\{\mathbf{V}_{s+1}^{(k)}{}^{\top}\left(\sum_{g=1}^{G} w_{(g)}^{(-k)}\mathbf{S}_{(g)}^{(k)}\mathbf{V}_s^{(k)}\mathbf{V}_s^{(k)\top}\mathbf{S}_{(g)}^{(k)}\right)\mathbf{V}_{s+1}^{(k)}\right\}\\
&= \operatorname{tr}\left(\mathbf{A}_s\mathbf{A}_{s+1}\right).
\end{aligned}$$

From the positivity of both sides of the inequality, it holds that

$$f_k(\mathbf{V}_s^{(k)})^2 \leq \left[\operatorname{tr}\left(\mathbf{A}_s\mathbf{A}_{s+1}\right)\right]^2.$$

In addition, from the Cauchy–Schwarz inequality, we have

$$\begin{aligned}
f_k(\mathbf{V}_s^{(k)})f_k(\mathbf{V}_{s+1}^{(k)}) &= \operatorname{tr}\left(\mathbf{A}_s\mathbf{A}_s\right)\operatorname{tr}\left(\mathbf{A}_{s+1}\mathbf{A}_{s+1}\right)\\
&\geq \left[\operatorname{tr}\left(\mathbf{A}_s\mathbf{A}_{s+1}\right)\right]^2.
\end{aligned}$$

Thus,

$$f_k(\mathbf{V}_s^{(k)})f_k(\mathbf{V}_{s+1}^{(k)}) \geq \left[\operatorname{tr}\left(\mathbf{A}_s\mathbf{A}_{s+1}\right)\right]^2 \geq f_k(\mathbf{V}_s^{(k)})^2.$$

Then, we have obtained $f_k(\mathbf{V}_s^{(k)})^2 \leq f_k(\mathbf{V}_s^{(k)})f_k(\mathbf{V}_{s+1}^{(k)})$. By dividing both sides of the inequality by $f_k(\mathbf{V}_s^{(k)})$, we obtain the inequality $f_k(\mathbf{V}_s^{(k)}) \leq f_k(\mathbf{V}_{s+1}^{(k)})$. □

Acknowledgments

We thank the reviewer for helpful comments and constructive suggestions. S.K. was supported by JSPS KAKENHI grants JP19K11854 and JP20H02227, and MEXT KAKENHI grants JP16H06429, JP16K21723, and JP16H06430.

References

Allen, G. (2012). Sparse higher-order principal components analysis. In *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics* (pp. 27–36).

Badeau, R., & Boyer, R. (2008). Fast multilinear singular value decomposition for structured tensors. *SIAM Journal on Matrix Analysis and Applications*, *30*(3), 1008–1021.

Bensmail, H., & Celeux, G. (1996). Regularized gaussian discriminant analysis through eigenvalue decomposition. *Journal of the American Statistical Association*, *91*(436), 1743–1748.

Boik, R. J. (2002). Spectral models for covariance matrices. *Biometrika*, *89*(1), 159–182.

Carroll, J. D., & Chang, J.-J. (1970). Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika*, *35*(3), 283–319.

Flury, B. N. (1984). Common principal components in K groups. *Journal of the American Statistical Association*, *79*(388), 892–898.

Flury, B. N. (1986). Asymptotic theory for common principal component analysis. *Annals of Statistics*, *14*(2), 418–430.

Flury, B. N. (1988). *Common principal components and related multivariate models*. New York: Wiley.

Flury, B. N., & Gautschi, W. (1986). An algorithm for simultaneous orthogonal transformation of several positive definite symmetric matrices to nearly diagonal form. *SIAM Journal on Scientific and Statistical Computing*, *7*(1), 169–184.

Greenewald, K., Zhou, S., & Hero III, A. (2019). Tensor graphical Lasso (TeraLasso). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *81*(5), 901–931.

Harshman, R. A. (1970). Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis. *UCLA Working Papers in Phonetics*, *16*(1), 84.

Harville, D. A. (1998). *Matrix algebra from a statistician's perspective*. New York: Springer-Verlag.

Jolliffe, I. (2002). *Principal component analysis*. New York: Springer-Verlag.

Kermoal, J. P., Schumacher, L., Pedersen, K. I., Mogensen, P. E., & Frederiksen, F. (2002). A stochastic MIMO radio channel model with experimental validation. *IEEE Journal on Selected Areas in Communications*, *20*(6), 1211–1226.

Kiers, H. A. (2000). Towards a standardized notation and terminology in multiway analysis. *Journal of Chemometrics*, *14*(3), 105–122.

Kolda, T. G., & Bader, B. W. (2009). Tensor decompositions and applications. *SIAM Review*, *51*(3), 455–500.

Lai, Z., Xu, Y., Chen, Q., Yang, J., & Zhang, D. (2014). Multilinear sparse principal component analysis. *IEEE Transactions on Neural Networks and Learning Systems*, *25*(10), 1942–1950.

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, *86*(11), 2278–2324.

Lu, H., Plataniotis, K. N., & Venetsanopoulos, A. N. (2008). MPCA: Multilinear principal component analysis of tensor objects. *IEEE Transactions on Neural Networks*, *19*(1), 18–39.

Manly, B. F. J., & Rayner, J. C. W. (1987). The comparison of sample covariance matrices using likelihood ratio tests. *Biometrika*, *74*(4), 841–847.

Martinez, A., & Benavente., R. (1998). *The AR face database* (CVC Technical Report 24).

Martinez, A. M., & Kak, A. C. (2001). PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *23*(2), 228–233.

Park, H., & Konishi, S. (2020). Sparse common component analysis for multiple high-dimensional datasets via noncentered principal component analysis. *Statistical Papers*, *61*, 2283–2311.

Pearson, K. (1901). LIII. On lines and planes of closest fit to systems of points in space. *London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, *2*(11), 559–572.

Pourahmadi, M., Daniels, M. J., & Park, T. (2007). Simultaneous modelling of the Cholesky decomposition of several covariance matrices. *Journal of Multivariate Analysis*, *98*(3), 568–587.

R Core Team (2019). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing.

Wang, H., Banerjee, A., & Boley, D. (2011). Common component analysis for multiple covariance matrices. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 956–964). New York: ACM.

Wang, S., Sun, M., Chen, Y., Pang, E., & Zhou, C. (2012). STPCA: Sparse tensor principal component analysis for feature extraction. In *Proceedings of the 21st International Conference on Pattern Recognition* (pp. 2278–2281). Piscataway, NJ: IEEE.

Werner, K., Jansson, M., & Stoica, P. (2008). On estimation of covariance matrices with Kronecker product structure. *IEEE Transactions on Signal Processing*, *56*(2), 478–491.

Yu, K., Bengtsson, M., Ottersten, B., McNamara, D., Karlsson, P., & Beach, M. (2004). Modeling of wide-band MIMO radio channels based on NLOS indoor measurements. *IEEE Transactions on Vehicular Technology*, *53*(3), 655–665.

Received November 20, 2020; accepted April 27, 2021.