# Clustering of multivariate binary data with dimension reduction via $L_1$-regularized likelihood maximization

Michio Yamamoto

Kyoto University Graduate School of Medicine
54 Kawahara-cho, Shogoin, Sakyo-ku, Kyoto 606-8507, Japan
Tel.: +81-75-751-4755
Fax: +81-75-751-4732
E-mail: michyama@kuhp.kyoto-u.ac.jp

Kenichi Hayashi

Osaka University Graduate School of Medicine
2-2 Yamadaoka, Suita, Osaka 565-0871, Japan
Tel.: +81-6-6879-3597
Fax: +81-6-6879-3598
E-mail: kenichi@medstat.med.osaka-u.ac.jp

Address correspondence to Michio Yamamoto, Department of Biomedical Statistics and Bioinformatics, Kyoto University Graduate School of Medicine, Kyoto 606-8507, Japan (e-mail: michyama@kuhp.kyoto-u.ac.jp).

## Abstract

Clustering methods with dimension reduction have been receiving considerable wide interest in statistics lately and a lot of methods to simultaneously perform clustering and dimension reduction have been proposed. This work presents a novel procedure for simultaneously determining the optimal cluster structure for multivariate binary data and the subspace to represent that cluster structure. The method is based on a finite mixture model of multivariate Bernoulli distributions, and each component is assumed to have a low-dimensional representation of the cluster structure. This method can be considered an extension of the traditional latent class analysis. Sparsity is introduced to the loading values, which produces the low-dimensional subspace, for enhanced interpretability and more stable extraction of the subspace. An EM-based algorithm is developed to efficiently solve the proposed optimization problem. We demonstrate the effectiveness of the proposed method by applying it to a simulation study and real datasets.

Key words: binary data, clustering, dimension reduction, EM algorithm, latent class analysis, sparsity

# 1 Introduction

Binary data are commonly observed and analyzed in many application fields: behavioral and social research, biosciences, document classification, and inference on binary images. For example, Ekholm et al. [1] analyzed biomedical data including five unequally spaced binary self-assessment measurements of arthritis and obesity data on the presence or absence of obesity in five cohorts of children. Also, the binarized data of the MovieLens 100K and the Netflix dataset, which are popular datasets for collaborative filtering tasks, have been analyzed by Kozma et al. [2]. One of the purposes of analyzing binary data, as well as continuous data, is the partitioning of objects which have binary features into several unpredetermined homogeneous groups (clusters). For clustering of objects with many variables, it is quite important to know if some of the variables do not contribute much to the structure of clusters because the inclusion of redundant information can reduce the performance of the cluster analysis [3]. Also, a lower-dimensional (say two or three dimensional) representation of the cluster structure, based on the most significant information, is very useful for evaluating and interpreting the results of the cluster analysis [4].

Hence, what is needed is a procedure that constructs a low-dimensional representation of the multivariate binary data, such that the cluster structure in the data is maximally revealed. For this purpose, researchers often carry out a preliminary dimension reduction technique (e.g., [5–10]). Among the references, [5] and [6] developed principal component analysis (PCA) models for binary data, while the other references have developed more general PCA models to handle exponential family data. Cluster analysis is then performed on the object scores on the first few principal components. Although it is easy to implement, this two-step sequential approach, also called the tandem approach, provides no assurance that the components extracted in the first step are optimal for the subsequent cluster analysis, because the two steps are implemented separately by optimizing a different loss function [4, 11–15]. For multivariate continuous data, instead of the two-step tandem clustering procedure, several methods that simultaneously perform cluster analysis and dimension reduction have been proposed [4, 13, 15–17].

On the other hand, for multivariate binary data, a few methods can conduct the analysis for simultaneously obtaining a cluster structure and a subspace for the cluster structure. Patrikainen and Mannila [18] have developed a subspace clustering method of binary data that can be used in high-dimensional settings. Cagnone and Viroli [19] have proposed a factor mixture analysis model for multivariate binary data, in which latent variables are distributed as a finite mixture of multivariate Gaussian distributions.

In general, there are two types of clustering techniques with finding subspaces: one intends to find a subspace that is common to all clusters [14], while the other aims to find a subspace specific to each group [20]. These two techniques can be used for different purposes. The former has a strong point in helping researchers to understand the configuration of objects and cluster centers in a single low-dimensional space. We need this technique if we want to analyze the data at hand using a component-based approach like the ordinary factor analysis and principal component analysis. The illustration shown in [4] is useful for understanding how to analyze the data using the common subspace clustering. On the other hand, the latter approach is needed for analyzing the data based on the assumption that the data points could be drawn from multiple subspaces. For example, a video sequence could contain several moving objects, and different subspaces might be needed to describe the motion of different objects in the scene [20]. In this paper, we focus on the common subspace clustering.

In the related works to the subspace clustering, there are several works on the problem of multi-task learning in which multiple tasks share a low-dimensional subspace. In the multi-task problem, parameters to be estimated are assumed to share some common structure in the tasks. For example, parameters are devided into two parts: one is common to all tasks and another is specific to each task [21]. Also, [22] assumes that tasks' structure is summarized by a positive definite matrix which is linked to the covariance matrix between the tasks. For supervised learning, [23] uses the formulation in which tasks share a linear low-dimensional subspace, and [24] proposes an optimization problem regularized by the projection distance of task-related parameters from the manifold shared by all tasks. In addition, for semi-supervised learning, there are some works that formulate the subspace shared by multiple tasks [25, 26].

As described above, Patrikainen and Mannila's [18] method allows for obtaining a cluster structure and a subspace for the cluster structure simultaneously. However, their method is rather cluster-specific subspace clustering. In addition, in the past few decades, because of technical advances in storing and processing data, we can obtain a large dataset that includes a large number of variables. Thus, we need to take into account such high-dimensional data. Cagnone and Viroli's [19] method, which is a common subspace clustering technique, cannot be used for a high-dimensional setting straightforwardly and may need strict restrictions for their parameters because of the identifiability problem.

Thus, we propose a new method to simultaneously find a cluster structure of multivariate binary data and an optimal low-dimensional space for clustering. The proposed model is based on the framework of latent class analysis (LCA) [27], which is used not only for analyzing the relation between categorical variables and discrete latent factors but for clustering objects with categorical features (e.g., [28]). Furthermore, our proposed method can deal with high-dimensional data.

The remainder of this paper is structured as follows. In Section 2, we introduce a new method to cluster multivariate binary data with dimension reduction. Section 3 describes an algorithm for the proposed optimization problem. Section 4 is devoted to studying the working of the clustering method using artificial and real data examples. Finally, we sum up our findings and set out directions for future expansion in Section 5.

## 2 Proposed method

Let $\tilde{\boldsymbol{y}} = (\tilde{y}_1, \dots, \tilde{y}_D)'$ be a random vector of $D$ binary variables. Suppose there are $K$ latent (unobservable) classes in a population and let $\tilde{u}_k$, $k = 1, \dots, K$, be an allocation variable that takes "1" if an observation belongs to class $k$, and "0" otherwise. We write $\tilde{\boldsymbol{u}} = (\tilde{u}_1, \dots, \tilde{u}_K)'$. We assume that the allocation variable follows a multinomial distribution, i.e., the probability that $\tilde{\boldsymbol{u}}$ takes the value $\boldsymbol{u} = (u_1, \dots, u_K)'$ is

$$f(\tilde{\boldsymbol{u}} = \boldsymbol{u}) = \prod_{k=1}^{K} \xi_k^{u_k},$$

where $\xi_k = \Pr(\tilde{u}_1 = 0, \dots, \tilde{u}_k = 1, \dots, \tilde{u}_K = 0)$.

Given that an observation is in the $k$th latent class, the probability that the random vector $\tilde{\boldsymbol{y}}$ takes the value $\boldsymbol{y} = (y_1, \dots, y_D)'$, where each $y_d$ takes 0 or 1, is represented as $\Pr(\tilde{\boldsymbol{y}} = \boldsymbol{y} \mid \tilde{u}_k = 1)$. The unconditional probability of the response $\boldsymbol{y}$ when we do not know the latent class of the observation is

$$\Pr(\tilde{\boldsymbol{y}} = \boldsymbol{y}) = \sum_{k=1}^{K} \xi_k \Pr(\tilde{\boldsymbol{y}} = \boldsymbol{y} \mid \tilde{u}_k = 1). \tag{2.1}$$

Here, we need to specify how the probability $\Pr(\tilde{\boldsymbol{y}} = \boldsymbol{y} \mid \tilde{u}_k = 1)$ depends on parameters. We postulate that, given the latent class to which an observation belongs, the responses on the binary variables are independent:

$$\Pr(\tilde{\boldsymbol{y}} = \boldsymbol{y} \mid \tilde{u}_k = 1) = \prod_{d=1}^{D} \Pr(\tilde{y}_d \mid \tilde{u}_k = 1). \tag{2.2}$$

This assumption of *conditional independence* has been widely used in latent class modeling in sociology [29], and is directly analogous to the assumption in the factor analysis model that observed variables are conditionally independent given the factors [27].

Finally, to specify the model completely, we need to specify a set of parameters that define the conditional probability of $\tilde{\boldsymbol{y}}$, with the value of $\tilde{\boldsymbol{u}}$ given. Suppose that $\tilde{\boldsymbol{y}}_1, \ldots, \tilde{\boldsymbol{y}}_N$ are mutually independent random variables that have the same distribution as $\tilde{\boldsymbol{y}}$, and the entries of $\mathbf{Y} = (y_{nd})$ are those realizations. We assume that, given the class $k$, $\tilde{y}_d$ follows the Bernoulli distribution with success probability $\pi_{kd}$. For the traditional LCA [27], we consider a parameter vector $\boldsymbol{\theta}_k = (\theta_{k1}, \ldots, \theta_{kD})'$, where $\theta_{kd}$ is the logit transformation of $\pi_{kd}$. We define the inverse logit transformation $\pi(\theta) = \{1 + \exp(-\theta))\}^{-1}$. The success probabilities can be represented using the canonical parameters $\theta_{kd}$ as $\pi_{kd} = \pi(\theta_{kd})$. Let $\tilde{y}_{nd}$ be the $d$th element of $\tilde{\boldsymbol{y}}_n$. The individual data-generating probability given the class then becomes

$$\begin{aligned}
\Pr(\tilde{y}_{nd} = y_{nd} \mid \tilde{u}_k = 1) &= \Pr(\tilde{y}_{nd} = y_{nd} \mid \tilde{u}_k = 1, \theta_{kd}) \\
&= \pi(\theta_{kd})^{y_{nd}} \{1 - \pi(\theta_{kd})\}^{1-y_{nd}} \\
&= \pi(q_{nd}\theta_{kd}),
\end{aligned}$$

with $q_{nd} = 2y_{nd} - 1$ since $\pi(-\theta) = 1 - \pi(\theta)$. Then, these representations lead to the compact form of the log likelihood as

$$\sum_{n=1}^{N} \log \left( \sum_{k=1}^{K} \xi_k \prod_{d=1}^{D} \pi(q_{nd}\theta_{kd}) \right).$$

We aim to obtain a low-dimensional representation of binary data in which the true cluster structure exists. Thus, we assume that canonical parameter $\theta_{kd}$ has a low-rank representation as follows:

$$\theta_{kd} = \mu_d + \boldsymbol{f}_k' \boldsymbol{a}_d, \tag{2.3}$$

where $\mu_d \in \mathbb{R}$, and for some positive integer $L$, $\boldsymbol{f}_k \in \mathbb{R}^L$ and $\boldsymbol{a}_d \in \mathbb{R}^L$. Here, $\mu_d$, $\boldsymbol{f}_k$, and $\boldsymbol{a}_d$ denote a centroid for the $d$th variable, a component score of the $k$th cluster, and a loading value for the $d$th variable, respectively. We write $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_K)'$, $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_D)'$, $\mathbf{F} = (\boldsymbol{f}_1, \ldots, \boldsymbol{f}_K)'$, and $\mathbf{A} = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_D)'$. To guarantee the determination of the decomposition for $\mathbf{F}$ and $\mathbf{A}$, we require that $\mathbf{F}$ has orthonormal columns. Then the log likelihood can be written as

$$\ell(\boldsymbol{\xi}, \boldsymbol{\mu}, \mathbf{F}, \mathbf{A}) = \sum_{n=1}^{N} \log \left( \sum_{k=1}^{K} \xi_k \prod_{d=1}^{D} \pi(q_{nd}(\mu_d + \boldsymbol{f}_k' \boldsymbol{a}_d)) \right). \tag{2.4}$$

Here, to deal with the high-dimensional problem, we assume that most of the elements of the true $\mathbf{A}$ are exactly zero. A sparse loading matrix implies variable selection in cluster analysis. That is, variables with non-zero loadings can be considered to contribute to a cluster structure in a low-dimensional space, whereas variables with zero loadings have no effect on the cluster structure. Furthermore, the introduction of sparsity of the loading coefficients through the representation (2.3) of the canonical parameters for a Bernoulli distribution will benefit the generalization of learned models. Thus, we propose to perform variable selection using the penalized likelihood with sparsity-inducing penalties. If $K = 1$ and $\boldsymbol{f}_k$ is observable, Eq. (2.4) is the log likelihood for $D$ logistic regression models. This connection with logistic regression suggests the use of the $L_1$ penalty to obtain a sparse loading matrix, as in the Lasso regression [30]. Specifically, consider the penalty

$$P_\lambda(\mathbf{A}) = \sum_{l=1}^{L} \lambda_l \|\breve{\boldsymbol{a}}_l\|_{L_1} = \lambda_1 \sum_{d=1}^{D} |a_{d1}| + \cdots + \lambda_L \sum_{d=1}^{D} |a_{dL}|,$$

where $\breve{\boldsymbol{a}}_l$ denotes the $l$th column of $\mathbf{A}$ and $\lambda_l$ is a regularization parameter. The choice of values for $\lambda_l$ will be discussed later. We obtain cluster components $\boldsymbol{\xi}$, $\boldsymbol{\mu}$, and $\mathbf{F}$ and a sparse loading matrix $\mathbf{A}$ by maximizing the following penalized log likelihood:

$$S(\boldsymbol{\xi}, \boldsymbol{\mu}, \mathbf{F}, \mathbf{A}) = \ell(\boldsymbol{\xi}, \boldsymbol{\mu}, \mathbf{F}, \mathbf{A}) - N \cdot P_\lambda(\mathbf{A}). \tag{2.5}$$

We call this procedure the clustering of binary data with reducing the dimensionality (CLUSBIRD). We can interpret penalized maximization as the device for generating a suitable optimization function, but not a realistic representation of the actual data-generating process. Thus, in this sense, the conditional independence given the latent class for obtaining the likelihood in Eq. (2.4) is assumed. A computational algorithm for solving the maximization problem is presented in the next section.

The effectiveness of the introduction of sparsity is illustrated in Figure 1 using a rank-two model (i.e., $L = 2$). The details of the setting will be presented in Section 4. While the regularized model can recover the original loading vector efficiently under the sparsity assumption, the unregularized model gives more noisy results. In the context of the ordinary factor analysis model, a sparse structure for the loading matrix provides an easy interpretation of the result, whereas it is difficult to interpret the relation between variables and factors if the loading matrix has no sparse structure. Browne [31] provides an excellent overview of the sparsity and rotation techniques which aim to obtain a sparse structure. In addition, Hirose and Yamamoto [32] discuss the sparsity problem in the factor analysis model, although their model aims to estimate the relation between continuous variables and continuous factors, and also cannot provide clusters of objects. Similar to the ordinary factor analysis model, noisy loading values may lead to difficulty in the interpretation of the result in our model. Thus, for the proposed model, sparse loading values offer an advantage.

Same as many mixture models, the proposed model also have identifiability problems on relabeling components, overfitting and generic identifiability [33]. Specifically, on the generic identifiability problem, it is known that in general the mixtures of binomial distributions are not identifiable (See for details [33] and references therein). In Section 4, we will evaluate the identifiability problem using artificial data.

# 3 Optimization Algorithm

As is often the case, we apply the EM algorithm [34] to solve the maximization problem (2.5). Let $\mathbf{U} = (u_{nk})$ be $N$ realizations of mutually independent random variable $\tilde{\boldsymbol{u}}$. In addition, denote the conditional probability (2.2) by $p_k(\boldsymbol{y} \mid \boldsymbol{\theta}_k)$. Then, the complete-data likelihood can be written as follows:

$$L^C(\mathbf{Y}, \mathbf{U} \mid \boldsymbol{\xi}, \boldsymbol{\mu}, \mathbf{F}, \mathbf{A}) = \prod_{n=1}^{N} \left\{ \prod_{k=1}^{K} p_k(\boldsymbol{y}_n \mid \boldsymbol{\theta}_k)^{u_k} \prod_{k=1}^{K} \xi_k^{u_k} \right\}.$$

As described in the previous section, we aim to obtain the sparse loading matrix $\mathbf{A}$; therefore, the penalty term for sparsity should be introduced. Thus, the complete-data log likelihood with the penalty is

$$\begin{aligned} &\ell^C(\mathbf{Y}, \mathbf{U} \mid \boldsymbol{\xi}, \boldsymbol{\mu}, \mathbf{F}, \mathbf{A}) \\ &\quad = \sum_{n=1}^{N}\sum_{k=1}^{K} u_{nk} \log p_k(\boldsymbol{y}_n \mid \boldsymbol{\theta}_k) + \sum_{n=1}^{N}\sum_{k=1}^{K} u_{nk} \log \xi_k - N \cdot P_\lambda(\mathbf{A}). \end{aligned} \tag{3.1}$$

The EM algorithm consists of a step maximizing the conditional expectation of the complete-data log-likelihood function (3.1) given the observable data $\mathbf{Y}$ and a set of parameters, $\{\boldsymbol{\xi}^{(t)}, \boldsymbol{\mu}^{(t)}, \mathbf{F}^{(t)}, \mathbf{A}^{(t)}\}$. Here, $\boldsymbol{\xi}^{(t)}$ denotes the value of $\boldsymbol{\xi}$ at the $t$th step in the algorithm, and this notation is applied to other parameters. From the above formulation, we can see that the penalized complete-data log likelihood (3.1) is a linear function with respect to values of $u_{nk}$. Thus, to obtain the conditional expected value of $\ell^C$, we only have to replace $u_{nk}$ with its conditional expectation,

$$\begin{aligned} u_{nk}^* &:= E\left[u_{nk} \mid \mathbf{Y}; \boldsymbol{\xi}^{(t)}, \boldsymbol{\mu}^{(t)}, \mathbf{F}^{(t)}, \mathbf{A}^{(t)}\right] \\ &= \frac{\xi_k^{(t)} p_k(\boldsymbol{y}_n \mid \boldsymbol{\theta}_{\boldsymbol{k}}^{(t)})}{\sum_{k=1}^{K} \xi_k^{(t)} p_k(\boldsymbol{y}_n \mid \boldsymbol{\theta}_{\boldsymbol{k}}^{(t)})}, \end{aligned} \tag{3.2}$$

where $\boldsymbol{\theta}_k^{(t)} = (\theta_{k1}^{(t)}, \ldots, \theta_{kD}^{(t)})'$, $k = 1, \ldots, K$, is obtained through Eq. (2.3) using $\{\boldsymbol{\xi}^{(t)}, \boldsymbol{\mu}^{(t)}, \mathbf{F}^{(t)}, \mathbf{A}^{(t)}\}$.

Thus, the conditional expectation of the complete-data log likelihood is as follows:

$$
\begin{aligned}
&Q(\boldsymbol{\xi}, \boldsymbol{\mu}, \mathbf{F}, \mathbf{A} \mid \boldsymbol{\xi}^{(t)}, \boldsymbol{\mu}^{(t)}, \mathbf{F}^{(t)}, \mathbf{A}^{(t)}) \\
&\qquad = E\left[\ell^C \mid \mathbf{Y}; \boldsymbol{\xi}^{(t)}, \boldsymbol{\mu}^{(t)}, \mathbf{F}^{(t)}, \mathbf{A}^{(t)}\right] \\
&\qquad = \sum_{n=1}^{N} \sum_{k=1}^{K} u_{nk}^{*} \log p_k(\boldsymbol{y}_n \mid \boldsymbol{\theta}_k) + \sum_{n=1}^{N} \sum_{k=1}^{K} u_{nk}^{*} \log \xi_k - N \cdot P_{\lambda}(\mathbf{A}).
\end{aligned}
$$

In the M-step of the EM algorithm, we consider the following maximization problem

$$
(\hat{\boldsymbol{\xi}}, \hat{\boldsymbol{\mu}}, \hat{\mathbf{F}}, \hat{\mathbf{A}}) = \underset{\boldsymbol{\xi}, \boldsymbol{\mu}, \mathbf{F}, \mathbf{A}}{\operatorname{argmax}}\, Q(\boldsymbol{\xi}, \boldsymbol{\mu}, \mathbf{F}, \mathbf{A} \mid \boldsymbol{\xi}^{(t)}, \boldsymbol{\mu}^{(t)}, \mathbf{F}^{(t)}, \mathbf{A}^{(t)}). \tag{3.3}
$$

Same as the usual mixture models, the estimate of $\boldsymbol{\xi}$ can be obtained by

$$
\hat{\xi}_k = N^{-1} \sum_{n=1}^{N} u_{nk}^{*}, \quad \text{for } k = 1, \ldots, K-1, \tag{3.4}
$$

and $\hat{\xi}_K = 1 - \sum_{k=1}^{K-1} \hat{\xi}_k$.

Given the estimate of $\boldsymbol{\xi}$, the maximization problem in (3.3) with respect to $\boldsymbol{\mu}$, $\mathbf{F}$, and $\mathbf{A}$ is equivalent to the minimization of the following function:

$$
g(\boldsymbol{\mu}, \mathbf{F}, \mathbf{A}) = -\sum_{n=1}^{N} \sum_{k=1}^{K} u_{nk}^{*} \log p_k(\boldsymbol{y}_n \mid \boldsymbol{\theta}_k) + N \cdot P_{\lambda}(\mathbf{A}). \tag{3.5}
$$

Here, the function $g$ in (3.5) is non-quadratic. Then, instead of directly dealing with the non-quadratic function $g$, we minimize a surrogate function, called the majorizing function [35], to solve the minimization problem of a quadratic function. In the majorization algorithm, a suitably defined quadratic upper bound of (3.5) is minimized, which provides optimal values for the actual function $m$. A function $h(x \mid y)$ is said to majorize a function $m(x)$ at $y$ if

$$
h(x \mid y) \geq m(x) \quad \text{for all } x \quad \text{and} \quad h(y \mid y) = m(y).
$$

In the geometrical view, the function surface $h(x \mid y)$ lies above the function $m(x)$ and is tangent to it at the point $y$; therefore $h(x \mid y)$ becomes an upper bound of $m(x)$. To minimize $m(x)$, the majorization algorithm decreases the objective function $m(x)$ in each step and is guaranteed to converge to a local minimum of $m(x)$. When applying the majorization algorithm, the majorizing function $h(x \mid y)$ is chosen so that it is easier to minimize than the original objective function $m(x)$. The study by Hunter and Lange [35] can be referred for an introductory description of the majorization algorithm.

To find a suitable majorizing function of (3.5), we consider the first term of (3.5). Note that, for a given point $y$,

$$-\log \pi(x) \leq -\log \pi(y) - \{1 - \pi(y)\}(x - y) + \frac{1}{8}(x - y)^2, \tag{3.6}$$

and the equality holds when $x = y$ [36, 37]. This equation provides quadratic upper bounds for the first term of (3.5) at the tangent point $y$. Thus we can apply the majorization algorithm for our problem.

We now present details of the majorization algorithm via the upper bound of $-\log \pi(x)$ in (3.6). By completing the square, Eq. (3.6) can be rewritten as

$$-\log \pi(x) \leq -\log \pi(y) + \frac{1}{8}\left[x - y - 4\{1 - \pi(y)\}\right]^2. \tag{3.7}$$

Substituting $x$ and $y$ with $q_{nd}\theta_{kd}$ and $q_{nd}\theta_{kd}^{(t)}$, respectively in (3.7) and using $q_{nd} = \pm 1$, we obtain

$$-\log \pi(q_{nd}\theta_{kd}) \leq -\log \pi(q_{nd}\theta_{kd}^{(t)}) + \frac{1}{8}(\theta_{kd} - z_{nkd}^{(t)})^2, \tag{3.8}$$

where

$$z_{nkd}^{(t)} = \theta_{kd}^{(t)} + 4q_{nd}\left\{1 - \pi(q_{nd}\theta_{kd}^{(t)})\right\}.$$

Thus, we obtain the following quadratic upper bound of the first term of (3.5):

$$\frac{1}{8}\sum_{n=1}^{N}\sum_{k=1}^{K} u_{nk}^{*} \sum_{d=1}^{D}(\theta_{kd} - z_{nkd}^{(t)})^2. \tag{3.9}$$

Eq. (3.9) then yields the following upper bound (up to a constant) of the criterion function $g(\boldsymbol{\mu}, \mathbf{F}, \mathbf{A})$ defined in (3.5):

$$\begin{aligned} h(\boldsymbol{\mu},&\mathbf{F}, \mathbf{A} \mid \boldsymbol{\mu}^{(t)}, \mathbf{F}^{(t)}, \mathbf{A}^{(t)}) \\ &= \frac{1}{8}\sum_{n=1}^{N}\sum_{k=1}^{K} u_{nk}^{*}\|\boldsymbol{z}_{nk}^{(t)} - (\boldsymbol{\mu} + \mathbf{A}\boldsymbol{f}_k)\|^2 + N \cdot P_\lambda(\mathbf{A}), \end{aligned} \tag{3.10}$$

where $\boldsymbol{z}_{nk}^{(t)} = (z_{nk1}^{(t)}, \ldots, z_{nkD}^{(t)})'$.

The majorizing function given in (3.10) is quadratic in each of $\boldsymbol{\mu}$, $\mathbf{F}$, and $\mathbf{A}$ when the other two are fixed, and thus alternating minimization of (3.10) with respect to $\boldsymbol{\mu}$ and $\mathbf{A}$ has closed-form solutions. We now drop the subscript $(t)$ for notational convenience. For fixed $\mathbf{F}$ and $\mathbf{A}$, set $\bar{z}_{kd} = N_k^{-1}\sum_{n=1}^{N} u_{nk}^{*} z_{nkd}$ where $N_k = \sum_{n=1}^{N} u_{nk}^{*}$, and write $\bar{\boldsymbol{z}}_k = (\bar{z}_{k1}, \ldots, \bar{z}_{kD})'$. Then the optimal $\hat{\boldsymbol{\mu}}$ is given by

$$\begin{aligned} \hat{\boldsymbol{\mu}} &= \underset{\boldsymbol{\mu}}{\operatorname{argmin}} \sum_{n=1}^{N}\sum_{k=1}^{K} u_{nk}^{*}\|\boldsymbol{z}_{nk} - (\boldsymbol{\mu} + \mathbf{A}\boldsymbol{f}_k)\|^2 \\ &= N^{-1}\sum_{k=1}^{K} N_k(\bar{\boldsymbol{z}}_k - \mathbf{A}\boldsymbol{f}_k). \end{aligned} \tag{3.11}$$

Optimization of $\mathbf{F}$ requires a numerical procedure because of its orthonormality. To update $\mathbf{F}$ for fixed $\boldsymbol{\mu}$ and $\mathbf{A}$, we apply the gradient projection (GP) algorithm with the orthonormal constraint [38, 39]. The only problem specific thing required for the GP algorithm is the gradient of (3.10) viewed as a function of $\mathbf{F}$. Let $\bar{z}_{kd}^{*} = N_k^{-1}\sum_{n=1}^{N} u_{nk}(z_{nkd} - \mu_d)$, and write $\bar{\mathbf{Z}}^{*} = (\bar{z}_{kd}^{*})$. Furthermore, let $\mathbf{N}$ be a $K \times K$ diagonal matrix where the $k$th diagonal element is $N_k$. Then, the gradient of $h$ at $\mathbf{F}$ is given as follows:

$$\boldsymbol{\Gamma} = \frac{\partial h}{\partial \mathbf{F}} = \frac{1}{4}\mathbf{N}(\mathbf{F}\mathbf{A}' - \bar{\mathbf{Z}}^*)\mathbf{A}. \tag{3.12}$$

Using $\boldsymbol{\Gamma}$ as the gradient in the GP algorithm with orthonormal constraint, we obtain the optimal $\hat{\mathbf{F}}$.

Finally, for fixed $\boldsymbol{\mu}$ and $\mathbf{F}$, the $dl$th element $a_{dl}$ of $\mathbf{A}$ is updated by solving the minimization problem in (3.10) directly. Let $v_{dl} = \sum_{n=1}^{N}\sum_{k=1}^{K} u_{nk}^*(z_{nkd} - \mu_d)f_{kl}$ and $w_{ll'} = \sum_{k=1}^{K} N_k f_{kl} f_{kl'}$. Then, up to a constant, the loss function with respect to $\mathbf{A}$ can be written as

$$h'(\mathbf{A}) = \frac{1}{8}\sum_{d=1}^{D}\sum_{l=1}^{L}\sum_{l'=1}^{L} w_{ll'}a_{dl}a_{dl'} - \frac{1}{4}\sum_{d=1}^{D}\sum_{l=1}^{L} v_{dl}a_{dl} + N\sum_{l=1}^{L}\lambda_l\sum_{d=1}^{D}|a_{dl}|.$$

Let $s_{dl} = \mathrm{sign}(a_{dl})$ for $a_{dl} \neq 0$, and $s_{dl} \in [-1, 1]$ for $a_{dl} = 0$. Thus, the subdifferential $\partial h'_{dl}(\mathbf{A})$ of $h'(\mathbf{A})$ at $a_{dl}$ is as follows:

$$\partial h'_{dl}(\mathbf{A}) = \left\{\frac{1}{4}\sum_{l'=1}^{L} w_{ll'}a_{dl'} - \frac{1}{4}v_{dl} + N\lambda_l s_{dl}\right\}. \tag{3.13}$$

Then, the optimal $\hat{a}_{dl}$ can be obtained by

$$\hat{a}_{dl} = \frac{1}{w_{ll}}\,\mathrm{sign}(c_{dl})\max(0, |c_{dl}| - 4N\lambda_l), \tag{3.14}$$

where $c_{dl} = -\sum_{l' \neq l} a_{dl'} + v_{dl}$.

The procedure of the proposed optimization algorithm is summarized in the following. The initial value of $\boldsymbol{\xi}$ is determined by the result of the ordinary $k$-means clustering for the data, and those of $\boldsymbol{\mu}$ and $\mathbf{A}$ are determined by the random numbers generated from the Gaussian distribution. The initial value of $\mathbf{F}$ is also determined by the Gaussian random numbers and then orthonormalized.

Clustering results are obtained by the posterior distribution of the allocation variable

**Algorithm 1** Optimization algorithm

```
1: Set $t = 1$ and initial values of $\boldsymbol{\xi}^{(1)}$, $\boldsymbol{\mu}^{(1)}$, $\mathbf{F}^{(1)}$, and $\mathbf{A}^{(1)}$.
2: Calculate the conditional expectation of $u_{nk}$ using (3.2).
3: Update $\boldsymbol{\xi}$ using (3.4) and set $\boldsymbol{\xi}^{(t+1)} = \hat{\boldsymbol{\xi}}$.
4: Update $\boldsymbol{\mu}$ using (3.11) and set $\boldsymbol{\mu}^{(t+1)} = \hat{\boldsymbol{\mu}}$.
5: Update $\mathbf{F}$ by the GP algorithm with the gradient $\boldsymbol{\Gamma}$ in (3.12) and set $\mathbf{F}^{(t+1)} = \hat{\mathbf{F}}$.
6: Update $\mathbf{A}$ using (3.14) and set $\mathbf{A}^{(t+1)} = \hat{\mathbf{A}}$.
7: If the penalized log likelihood (2.5) converges, then the algorithm finishes. If not,
   $t \leftarrow t + 1$ and go to the step 2.
```

$u_{nk}$ with estimated parameters, $\hat{\boldsymbol{\xi}}$, $\hat{\boldsymbol{\mu}}$, $\hat{\mathbf{F}}$, and $\hat{\mathbf{A}}$. That is, the estimated number $\hat{k}$ of the cluster to which the $n$th subject is assigned is given by

$$\hat{k} = \underset{k}{\operatorname{argmax}} \frac{\hat{\xi}_k p_k(\boldsymbol{y}_n \mid \hat{\boldsymbol{\theta}}_k)}{\sum_{l=1}^{K} \hat{\xi}_l p_l(\boldsymbol{y}_n \mid \hat{\boldsymbol{\theta}}_l)}$$

with $\hat{\theta}_{kd} = \hat{\mu}_d + \hat{\boldsymbol{f}}_k' \hat{\boldsymbol{a}}_d$ and $\hat{\boldsymbol{\theta}}_k = (\hat{\theta}_{k1}, \ldots, \hat{\theta}_{kD})'$.

Also, the individual component score $\mathbf{G} = (g_{nl})$ $(n = 1, \ldots, N; l = 1, \ldots, L)$, which expresses the low-dimensional configuration of an individual object $n$, can be obtained by the post hoc model with the estimated parameters. The detailed procedure is described in Appendix A.

Prior to applying the above algorithm, the value of the regularization parameters, $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_L)'$, should be determined. We use the cross-validation approach for mixture models. That is, we choose the one that maximizes the five-fold cross-validated likelihood. Although different parameters can be used for different component loading vectors, we consider using only a single regularization parameter $\lambda$ for all loadings. Also, we need to determine the number of components $L$. Since $K - 1$ components are sufficient to express the configuration of $K$ clusters [4, 15], we choose the value of $L$ in $\{1, \ldots, K - 1\}$ based on the cross-validated likelihood. In real data analyses in Section 4.2 and 4.3, we demonstrate this selection procedure.

# 4 Numerical examples

## 4.1 A Monte Carlo simulation

We conducted a simulation study to evaluate the performance of the proposed method, compared with tandem analysis (TA), in which sparse logistic principal component analysis (SLPCA) [6] is conducted, followed by the ordinary $k$-means clustering of estimated principal component scores. Also, on the performance of recovering a true cluster structure, we compared the proposed model with Bouguila's [40] feature weighting clustering technique (FW). Bouguila's model aims to estimate cluster assignments of objects with multivariate binary features, although the method cannot provide the component score for individual objects unlike the proposed model and TA. For FW, we used the uniform distribution as prior distributions of parameters.

The artificial data $\mathbf{Y}$ were generated through the CLUSBIRD model (2.1) with three clusters ($K = 3$) and two dimensional structure ($L = 2$). That is, an object $y_{nd}$ that was assigned to cluster $k$ was generated by $y_{nd} \sim Ber(\pi_{kd})$. To determine the value of $\pi_{kd}$, the values of $\boldsymbol{\mu}$, $\mathbf{F}$, and $\mathbf{A}$ were generated. We used a zero vector for $\boldsymbol{\mu}$. Each centroid $\boldsymbol{f}_k$ of clusters in the two-dimensional space were randomly generated so that the distance between two clusters was equal for all combinations of two clusters, and then the $\mathbf{F} = (\boldsymbol{f}_1, \boldsymbol{f}_2, \boldsymbol{f}_3)$ was orthonormalized. The loading matrix $\mathbf{A}$ was set at

$$\mathbf{A} = \begin{pmatrix} c \cdot \mathbf{1}_{D_1} & \mathbf{0}_{D_1} \\ \mathbf{0}_{D_1} & c \cdot \mathbf{1}_{D_1} \\ \mathbf{0}_{D_2} & \mathbf{0}_{D_2} \end{pmatrix},$$

where $\mathbf{1}_q$ and $\mathbf{0}_q$ denote $q$-vectors of ones and zeroes, respectively. Here, $c$ is a scalar whose value was determined based on sample size as described below. In this simulation study, we considered three factors: sample size ($N = 100,\ 300$), the number of variables ($D = 10,\ 1000$), and the proportion of informative variables on the cluster structure

($m = 0.5,\ 1.0$). Then, we set the value of $c$ was set at 2.5 for $D = 10$ and 0.5 for $D = 1000$. The number $D_1$ was calculated as $D_1 = \lfloor \frac{m}{2} D \rfloor$, where $\lfloor \cdot \rfloor$ denotes a floor function. Thus, based on the above structure of $\mathbf{A}$, $2D_1$ variables contributes the low-dimensional structure and $D_2 (= D - 2D_1)$ variables are random error variables. For each condition, we generated 50 replications, thus yielding $2 \times 2 \times 2 \times 50 = 400$ random samples in total.

We used the adjusted Rand index (ARI) [41] and the normalized mutual information measure (NMI) [42] to assess the recovery of cluster memberships. The ARI has a maximal value of 1 in the case of a perfect recovery of the underlying cluster structure, and a value of 0 in the case where the true and estimated class assignments coincide no more than would be expected by chance. The NMI is bounded range $[0, 1]$, in which a value of 0 indicates a purely independent label assignment and a value close to 1 indicates significant agreement.

In this study, we used 50 sets of random initial values of all parameters for the proposed model and FW, and 10 sets for SLPCA. Also, we used the parameter values of $K$ as its value, i.e., a value of 3 for the three methods and $L$ as its value, i.e., a value of 2 for the proposed model and TA. The values of tuning parameter $\lambda$ in the SLPCA model were determined by BIC as defined in [6]. To reduce computational burden, we selected the values of tuning parameters only in the first replication for each condition, and then used the values of the parameter obtained from the selection by BIC (for SLPCA) or cross-validation (for the proposed model) for the remaining replications.

Table 1 shows results of the ARI and NMI obtained from the three methods, along with the values of $D$, $m$, and $N$. Each figure shows the median values of the ARI or NMI for 50 replications under each condition. Regardless of the number of variables, the proposed method provided better or equivalent results than tandem analysis under all conditions. We can see that the recovery of the cluster structure became better when the sample size and/or the proportion of the informative variables increased. Also, compared with FW, the recoveries of the proposed method were superior or similar to those of FW in almost all conditions.

Next, we evaluated the local optima using the artificial data used in the above simulation. Here, two cases were considered: one was the data set consisting of 10 variables and the other was the data set consisting of 1000 variables. Also, the sample size for both cases was 300 and the proportion $m$ of informative variables on the cluster structure was 1.0. Figure 2 shows that trajectories of the values of the penalized log-likelihood functions in optimization processes of 200 different initial starts. We can see that for the case where $D = 10$, many initial starts attained the maximal solution. Actually, 162 out of 200 initial starts converged at the same value of the penalized log likelihood. On the other hand, for the case of $D = 1000$, only the two initial values attained the maximal solution. Thus, when a data set at hand is large, i.e., $D$ is large, it is recommended to use sufficiently large number of initial starts.

In addition, we evaluate the identifiability problem using the above result. For the case of $D = 10$, the solutions that had the same value of the penalized log likelihood provided the same partitioning and thus they had the same value of the ARI (0.902). Thus, in this case, the identifiability problem of the proposed model had no influence on the clustering performance. Also, we evaluated the performance of estimating the low-dimensional expression of the data. Figure 3 shows the plot of estimated component scores of individual objects for the solutions obtained by the proposed model with two different initial starts that provide the same value of the penalized log likelihood. We can see that the two configurations are identical. Thus, it can be concluded that the solutions with the same value of the penalized log likelihood provide the same partitioning and the same low-dimensional expression.

In the case where $D = 1000$, the two solutions, which attained the almost the same value (within the precision) of the penalized log likelihood, shown at the right panel in Figure 2, had the same partitioning except two objects. This may be due to the little difference in their values of the penalized log likelihoods of -204228.1 and -204226.5. The two solutions provided the same value of the ARI of 0.990, and thus the difference implies little impact on interpreting the cluster structure. In addition, Figure 4 shows the plot of estimated components scores of individuals for the two solutions. Although there is a small difference between the two configurations due to the difference in partitionings, we can interpret one low-dimensional expression in almost the same manner as the other one.

From the above evaluations, we conclude that the identifiability problem of the proposed model have little impact on analyzing the data in practice, as long as we use many initial values to maximize the penalized log-likelihood function and check their results.

### 4.2 Binary image classifications

Handwritten digit recognition has many application scenarios such as auto-mail classification according to zip code and signature recognition [40]. We used binary image data that were available from the well-known UCI database [43] which contains 5,620 objects. Each object represents one of the integers from 0 to 9, and we used images of 1, 2, 3, and 4, for which examples are shown in Figure 5. Each normalized bitmap includes a $32 \times 32$ matrix, i.e., a 1,024-dimensional binary vector, in which each element indicates one pixel with a value of white or black. Fifty objects for each number were selected and thus $50 \times 4 = 200$ objects were analyzed by the proposed method, tandem analysis (TA), and Bouguila's method (denoted as FW) with $K = 4$. For the proposed method, the value of a tuning parameter $\lambda$ was determined by the cross-validation and for the tandem approach, we used BIC.

First, for the proposed model, we selected the value of $L$ using cross-validation approach, which resulted in $L = 3$. And then, we implemented the proposed model and tandem analysis with $L = 3$ and Bouguila's method. The values of the ARI and NMI obtained by the three methods are shown in Table 2. The CLUSBIRD and FW model showed much the same clustering recovery and the results of the two method on the clustering recovery were better than that of the tandem analysis.

Next, to compare the performance of the proposed method with the tandem approach in terms of finding a suitable low-dimensional configuration for the cluster structure, the two method were applied to estimate component scores of individual objects with $L = 2$. Estimated component scores with clusters are shown in Figure 6. It can be seen that the proposed method provided a well-separated and compact low-dimensional cluster structure. On the other hand, tandem analysis provided crude recovery of the true cluster structure. From the low-dimensional expression, we can explore the characteristics of objects and the cluster structure graphically. For example, we can see the distance of a certain object from other clusters, which determines the characteristic of the object, and also we can grasp the shapes of clusters.

### 4.3 Population classification using single nucleotide polymorphism data

Association studies based on high-throughput single nucleotide polymorphism (SNP) data have become a popular way to detect genomic regions associated with complex human diseases. A crucial issue in association studies is population stratification detection [44], which is to determine whether a population is homogeneous or has hidden structures within it. With the presence of population stratification, a naive case-control approach that did not consider the stratification would yield biased results and, therefore, draw inaccurate scientific conclusions [45]. We used the SNP dataset available in the International HapMap project [46], filtering out those with minor allele frequencies greater than 0.01 and those missing genotype rates less than 0.05. The dataset consists of 3 different ethnic populations of 90 Asians (45 Han Chinese in Beijing, China; CHB and 45 Japanese in Tokyo, Japan; JPT), 60 Caucasians (Utah residents with ancestry from northern and western Europe; CEO), and 60 Africans (Yoruba in Ibadan, Nigeria; YRI). Here, we conducted the proposed method and tandem analysis to detect the three-subpopulation structure using the SNP data on the 210 subjects.

Since there were too many SNPs (2.2 million, 2.3 million, and 2.6 million SNPs for CHB-JPT, CEO, and YRI populations, respectively) to analyze those data, we had to select SNPs that were seen to be associated with detection of the subpopulation. First, using PLINK [47], we conducted three association analyses in which each population was considered as a case and the other two populations were control. Then, we obtained SNPs which had genome-controlled p-values less than 0.1%. All those SNPs were considered to be related to the differences among the three ethnic populations. After selecting SNPs with no missing values, we finally obtained 589 SNPs of 210 subjects.

We conducted the proposed CLUSBIRD method, tandem analysis, and Bouguila's method with $K = 3$ using the SNP data. A tuning parameter $\lambda$ was determined by BIC for tandem approach and the cross-validation for CLUSBIRD. First, using the proposed model, we selected the value of $L$ using the cross-validation approach, which resulted in $L = 2$. Then, we implemented CLUSBIRD and tandem analysis with $L = 2$. The results are shown in Figure 7. We can see that the proposed method recovered the true ethnic populations perfectly. In contrast, tandem analysis provided a crude recovery of the populations. The tandem analysis, SLPCA, provided a bit sparse estimation of loading values, where only a few SNPs had large loading values for the first component and many SNPs had low loading values for the second component. Although this sparse structure may provide easy interpretation for the estimated low-dimensional structure, the structure did not contain the true ethnic populations well. The proposed method provided a reasonably sparse structure of loading values. Actually, all SNPs used for this analysis had some relation to the detection of populations. Thus, it is reasonable that all SNPs had large loading values. In addition, for the proposed method, almost all SNPs had high loading values for one component, resulting in easy interpretation of the low-dimensional structure. In addition, for comparison, we conducted the Bouguila's model to the data. Then, the result of the ARI is 1.0, that is, the Bouguila's method can also recover the true cluster structure perfectly, though the method cannot provide the component scores of objects like CLUSBIRD and tandem analysis.

## 5 Conclusion

In this paper, we proposed a new procedure, called CLUSBIRD, for simultaneously finding the optimal cluster structure for multivariate binary objects and finding the subspace to represent the cluster structure. The proposed method can provide the weight for each binary variable, which indicates the contribution of the variable to the cluster structure. In general, tandem analysis for clustering objects with dimension reduction is likely to fail in finding the cluster structure. In fact, our numerical examples demonstrate the inability of tandem analysis to detect the cluster structure and subspace for the structure. Those examples also show that our proposed method can provide a better cluster structure than tandem analysis. Furthermore, from the examples, we found that our procedure can work well for data that had a mildly larger number of variables than the sample size.

The proposed model can be considered an extension of the ordinary latent class analysis (LCA) [27]. However, the ordinary LCA cannot provide loading values for variables and a low-dimensional structure. Also, LCA may not provide an appropriate estimation with the moderately high-dimensional dataset we used in the numerical examples. From this point of view, the proposed method can provide useful insight for researchers.

The proposed method can be extended to deal with various problems. For example, it is useful for the proposed model to deal with categorical variables, not just binary variables. In addition, the ordinary LCA model is ready for multi-group analysis, the analysis with covariates, and analysis of repeated measures data [29]. Using the formulation of LCA, the proposed model can also contain those features. Furthermore, we can develop a cluster-specific subspace clustering technique based on the CLUSBIRD model. These could be interesting topics for further research.

## Appendix A: Estimation of individual latent scores

To obtain individual component scores, $\mathbf{G} = (g_{nl})$, we propose a two-step approach. First, we estimate all parameters, $\boldsymbol{\mu}$, $\mathbf{A}$, $\mathbf{F}$, and $\boldsymbol{\xi}$, in the CLUSBIRD model. Then, we assume that a cluster structure of individuals is present in a low-dimensional space that is the same as that for the cluster center $\mathbf{F}$. That is, the estimated loading matrix $\hat{\mathbf{A}}$ and low-dimensional centroids $\hat{\boldsymbol{\mu}}$ also define the subspace for the individuals. Thus, we consider the following post hoc model. Suppose that $\tilde{y}_{nd}$ $(n = 1, \ldots, N;\ d = 1, \ldots, D)$ follows the Bernoulli distribution with success probability $\pi_{nd} = \pi(\theta_{nd})$, where $\theta_{nd}$ is the logit transformation of $\pi_{nd}$. In addition, we assume that the canonical parameter $\theta_{nd}$ has a low-rank representation

$$\theta_{nd} = \hat{\mu}_d + \boldsymbol{g}_n' \hat{\boldsymbol{a}}_d,$$

where $\boldsymbol{g}_n = (g_{n1}, \ldots, g_{nL})'$ with $\mathbf{G}'\mathbf{G} = \mathbf{I}_L$. Here, we write

$$S(\mathbf{G}) = \sum_{n=1}^{N} \sum_{d=1}^{D} \log \pi(q_{nd}(\hat{\mu}_d + \boldsymbol{g}_n' \hat{\boldsymbol{a}}_d)).$$

Then, we obtain individual component scores by maximizing $S(\mathbf{G})$ over $\mathbf{G}$. Similar to the solution of $\mathbf{F}$ in Section 3, the optimal $\mathbf{G}$ can be obtained using the GP algorithm.

## Acknowledgment

We thank the Editor and two anonymous reviewers for their constructive comments that helped to improve the quality of this article. This work was supported by Japan Society for the Promotion of Science (JSPS) KAKENHI Grand Number 26730016 and 24700276.

# References

[1] A. Ekholm, J.W. McDonald, and P.W.F. Smith. Association models for a multivariate binary response. *Biometrics*, 56:712–718, 2000.

[2] L. Kozma, A. Ilin, and T. Raiko. Binary principal component analysis in the netflix collaborative filtering task. In *Proceedings of 2009 IEEE International Workshop on Machine Learning for Signal Processing*, 2009.

[3] G.W. Milligan. Clustering validation: Results and implications for applied analysis. In P. Arabie, L.J. Hubert, and G. De Soete, editors, *Clustering and Classification*, pages 341–375. World Scientific Publishing, River Edge, 1996.

[4] M. Vichi and H.A.L. Kiers. Factorial k-means analysis for two-way data. *Computational Statistics & Data Analysis*, 37(1):49–64, 2001.

[5] A.I. Schein, L.K. Saul, and L.H. Ungar. A generalized linear model for principal component analysis of binary data. In C.M. Bishop and B.J. Frey, editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, volume 38, pages 14–21. Key West, Florida, 2003.

[6] S. Lee, J.Z. Huang, and J. Hu. Sparse logistic principal components analysis for binary data. *The Annals of Applied Statistics*, 4:1579–1601, 2010.

[7] I. Moustaki and M. Knott. Generalized latent trait models. *Psychometrika*, 65:391–411, 2000.

[8] M. Collins, S. Dasgupta, and R.E. Schapire. A generalization of principal component analysis to the exponential family. In T.G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advanced in Neural Information Processing System*, volume 14, pages 617–642. MIT Press, Cambridge, MA, 2002.

[9] J. Li and D. Tao. Simple exponential family pca. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)2010*, 2010.

[10] J. Li and D. Tao. Exponential family factors for bayesian factor analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 24:964–976, 2013.

[11] P. Arabie and L. Hubert. Cluster analysis in marketing research. In R. P. Bagozzi, editor, *Advanced methods of marketing research*, pages 160–189. Blackwell, Oxford, 1994.

[12] W. S. DeSarbo, K. Jedidi, K. Cool, and D. Schendel. Simultaneous multidimensional unfolding and cluster analysis: An investigation of strategic groups. *Marketing Letters*, 2(2):129–146, 1990.

[13] G. De Soete and J. D. Carroll. K-means clustering in a low-dimensional euclidean space. In E. Diday, Y. Lechevallier, M. Schader, P. Bertrand, and B. Burtschy, editors, *New approaches in classification and data analysis*, pages 212–219, Berlin, Heidelberg, 1994. Springer.

[14] M. E. Timmerman, E. Ceulemans, H. A. L. Kiers, and M. Vichi. Factorial and reduced k-means reconsidered. *Computational Statistics & Data Analysis*, 54:1858–1871, 2010.

[15] M. Yamamoto and H. Hwang. A general formulation of cluster analysis with dimension reduction and subspace separation. *Behaviormetrika*, 41:115–129, 2014.

[16] Z. Ghahramani and G.E. Hilton. The em algorithm for mixture of factor analyzers. Technical Report CRG-TR-96-1, Department of Computer Science, University of Toronto, Canada, 1997.

[17] R. Yoshida, T. Higuchi, and S. Imoto. A mixed factors model for dimension reduction and extraction of a group structure in gene expression data. In *Proceedings of*

*the 2004 IEEE Computational Systems Bioinformatics Conference*, pages 161–172, 2004.

[18] A. Patrikainen and H. Mannila. Subspace clustering of high-dimensional binary data - a probabilistic approach. In *Workshop on Clustering High Dimensional Data and its Applications, SIAM International Conference on Data Mining*, pages 57–65, 2004.

[19] S. Cagnone and C. Viroli. A factor mixture analysis model for multivariate binary data. *Statistical Modelling*, 12:257–277, 2012.

[20] R. Vidal. Subspace clustering. *Signal Processing Magazine*, 28:52–68, 2011.

[21] T. Evgeniou and M. Pontil. Regularized multi-task learning. In *KDD '04 Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 109–117. 2004.

[22] A. Argyriou, C.A. Micchelli, M. Pontil, and Y. Ying. A spectral regularization framework for multi-task structure learning. In *Advances in Newural Information Processing Systems 20*, pages 25–32. 2007.

[23] S. Ji, L. Tang, S. Yu, and J. Ye. A shared-subspace learning framework for multi-label classification. *ACM Transactions on Knowledge Discovery from Data*, 4(2):Article 8, 2010.

[24] A. Agarwal, H. Daumé III, and S. Gerber. Learning multiple tasks using manifold regularization. In *Proceedings of Conference on Neural Inforamtion Processing Systems (NIPS)*. 2010.

[25] R.K. Ando and T. Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853, 2005.

[26] Y. Luo, D. Tao, B. Geng, C. Xu, and S.J. Maybank. Manifold regularized multitask learning for semi-supervised multilabel image classification. *IEEE Transactions on Image Processing*, 22(2):523–536, 2013.

[27] M. Aitkin, D. Anderson, and J. Hinde. Statistical modeling of data on teaching styles. *Journal of the Royal Statistical Society, Series A*, 144:419–461, 1981.

[28] J. Magidson and J. K. Vermunt. Latent class models for clustering. *Canadian Journal of Marketing Research*, 20:37–44, 2002.

[29] L.M. Collins and S.T. Lanza. *Latent class and latent transition analysis with applications in the social, behavioral, and health sciences.* John Wiley & Sons, Inc., New Jersey, 2010.

[30] R.J. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1996.

[31] M. W. Browne. An overview of analytic rotation in exploratory factor analysis. *Multivariate Behavioral Research*, 36(1):111–150, 2001.

[32] K. Hirose and M. Yamamoto. Sparse estimation via nonconcave penalized likelihood in a factor analysis model. *Statistics and Computing*, in press.

[33] S. Frühwirth-Schnatter. *Finite mixture and Markov switching models.* Springer, New York, 2006.

[34] N.M. Dempster, A.P. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the em algorithm (with discussion). *Journal of the Royal Statistical Society B*, 39:1–38, 1977.

[35] D.R. Hunter and K. Lange. A tutorial on mm algorithms. *The American Statistician*, 58:30–37, 2004.

[36] T.S. Jaakkola and M.I. Jordan. Bayesian parameter estimation via variational methods. *Statistics and Computing*, 10:25–37, 2000.

[37] J. De Leeuw. Principal component analysis of binary data by iterated singular value decomposition. *Computational Statistics & Data Analysis*, 50:21–39, 2006.

[38] R. I. Jennrich. A simple general procedure for orthogonal rotation. *Psychometirka*, 66(2):289–306, 2001.

[39] R. I. Jennrich. A simple general procedure for oblique rotation. *Psychometirka*, 67(1):7–20, 2002.

[40] N. Bouguila. On multivariate binary data clustering and feature weighting. *Computational Statistics & Data Analysis*, 54:120–134, 2010.

[41] L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

[42] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, P09008, 2005.

[43] K. Bache and M. Lichman. UCI machine learning repository, 2013.

[44] K. Hao, C. Li, C. Rosenow, and W.H. Wong. Detect and adjust for population stratification in population-based association study using genomic control markers: An application of affymetrix genechip® human mapping 10k array. *European Journal of Human Genetics*, 12:1001–1006, 2004.

[45] W.J. Ewens and R.S. Spielman. The transmission/disequilibrium test: History, subdivision, and admixture. *The American Journal of Human Genetics*, 57:455–464, 1995.

[46] The International HapMap Consortium. A haplotype map of the human genome. *Nature*, 437:1299–1320, 2005.

[47] S. Purcell, B. Neale, K. Todd-Brown, L. Thomas, M.A.R. Ferreira, D. Bender, J. Maller, P. Sklar, P.I.W. de Bakker, M.J. Daly, and P.C. Sham. Plink: a toolset for whole-genome association and population-based linkage analysis. *American Journal of Human Genetics*, 81:559–575, 2007.

Table 1: Median values of adjusted Rand index (ARI) and normalized mutual information criterion (NMI) for 50 replications in each condition; $D$, $m$, and $N$ denotes the number of variables, the proportion of informative variables on the cluster structure, and sample size, respectively.

| Measure | $D$ | $m$ | $N$ | TA | FW | CLUSBIRD |
|---|---|---|---|---|---|---|
| ARI | 10 | 0.5 | 100 | 0.189 | 0.250 | 0.370 |
| | | | 300 | 0.354 | 0.418 | 0.450 |
| | | 1.0 | 100 | 0.740 | 0.770 | 0.797 |
| | | | 300 | 0.563 | 0.828 | 0.833 |
| | 1000 | 0.5 | 100 | 0.010 | 0.017 | 0.005 |
| | | | 300 | 0.025 | 0.275 | 0.488 |
| | | 1.0 | 100 | 0.021 | 0.075 | 0.000 |
| | | | 300 | 0.017 | 1.000 | 0.990 |
| NMI | 10 | 0.5 | 100 | 0.208 | 0.287 | 0.337 |
| | | | 300 | 0.326 | 0.397 | 0.410 |
| | | 1.0 | 100 | 0.700 | 0.734 | 0.759 |
| | | | 300 | 0.500 | 0.769 | 0.775 |
| | 1000 | 0.5 | 100 | 0.027 | 0.040 | 0.045 |
| | | | 300 | 0.029 | 0.269 | 0.564 |
| | | 1.0 | 100 | 0.043 | 0.098 | 0.025 |
| | | | 300 | 0.023 | 1.000 | 0.983 |

Table 2: The values of the ARI and NMI obtained by the three methods for binary image data.

| | TA | FW | CLUSBIRD |
|---|---|---|---|
| ARI | 0.593 | 0.896 | 0.883 |
| NMI | 0.610 | 0.881 | 0.863 |

## Figure captions

*Figure 1.* The results of analyzing an artificial dataset with $N = 100$, $D = 100$, $L = 2$, and $K = 4$; top, middle, and bottom panels show the true loadings, absolute values of loadings from the unregularized model, and absolute values of loadings from the regularized model, respectively; left and right panels show loadings for the first and second components, respectively; the penalty parameter was selected using the five-fold cross-validation.

*Figure 2.* Trajectories of the values of the penalized log-likelihood functions in optimization processes of 200 initial values; the left panel and the right panel show those for the cases where the numbers of variables are 10 and 1000, respectively; a red line indicates the best value of the penalized log likelihood in each case.

*Figure 3.* Plots of component scores estimated by CLUSBIRD using the two initial starts that provide the same value of the penalized log likelihood for the data set consisting of 10 variables; the colors and shapes of plotted points denote the true and estimated memberships, respectively.

*Figure 4.* Plots of component scores estimated by CLUSBIRD using the two initial starts that provide the same value of the penalized log likelihood for the data set consisting of 1000 variables; the colors and shapes denote the true and estimated memberships, respectively.

*Figure 5.* Examples of normalized bitmaps.

*Figure 6.* Plots of component scores estimated by CLUSBIRD and tandem analysis; in the plots, the number denotes the estimated cluster and the color denotes the true cluster.

*Figure 7.* Plots of component scores (left) and loading values (right) estimated by CLUSBIRD and tandem analysis; in the left panel, the colors and shapes denote the true and estimated memberships, respectively; loading values were scaled so that the value existed in $[-1, 1]$.

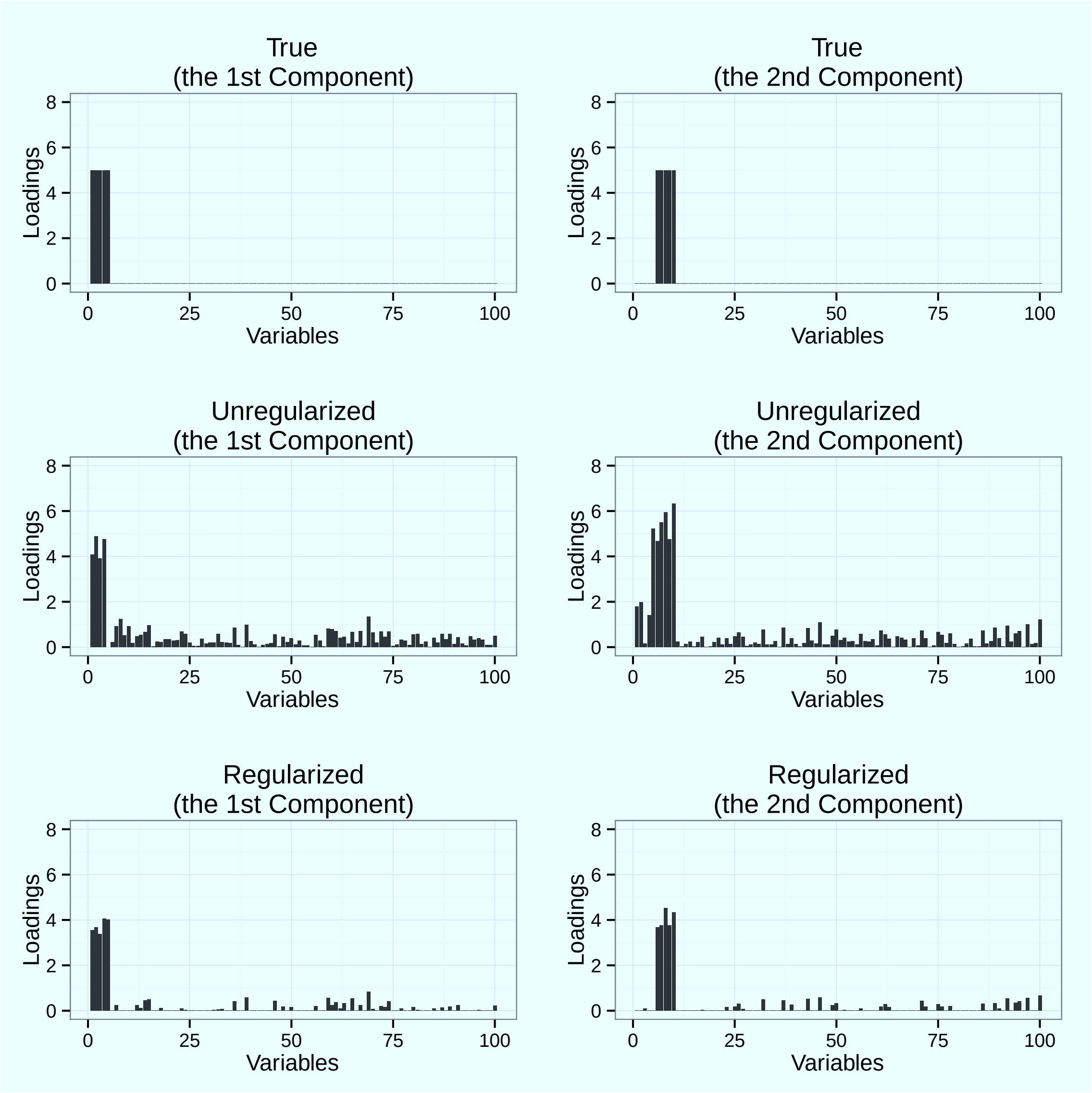


Figure 1: The results of analyzing an artificial dataset with $N = 100$, $D = 100$, $L = 2$, and $K = 4$; top, middle, and bottom panels show the true loadings, absolute values of loadings from the unregularized model, and absolute values of loadings from the regularized model, respectively; left and right panels show loadings for the first and second components, respectively; the penalty parameter was selected using the five-fold cross-validation.

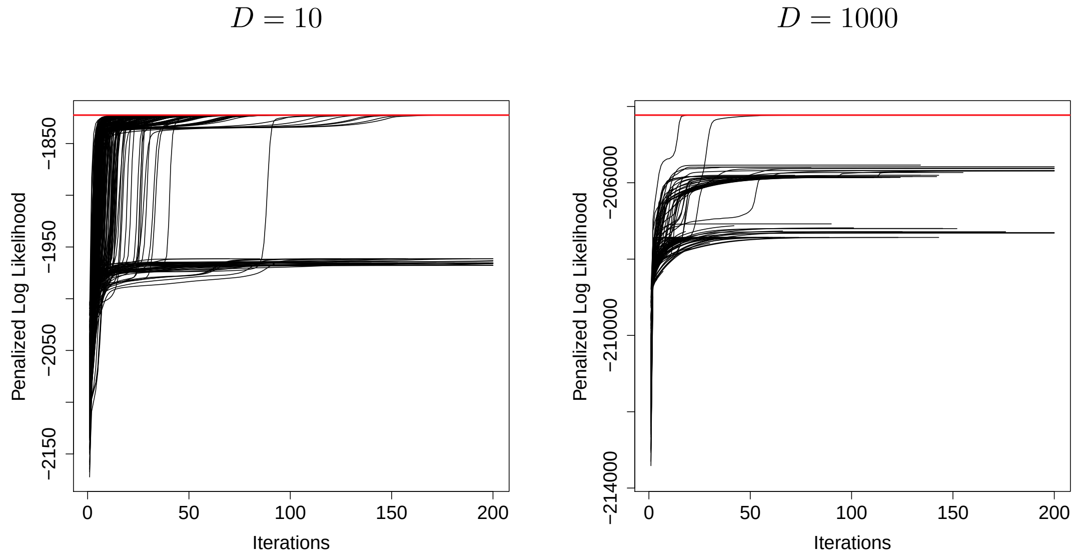


Figure 2: Trajectories of the values of the penalized log-likelihood functions in optimization processes of 200 initial values; the left panel and the right panel show those for the cases where the numbers of variables are 10 and 1000, respectively; a red line indicates the best value of the penalized log likelihood in each case.

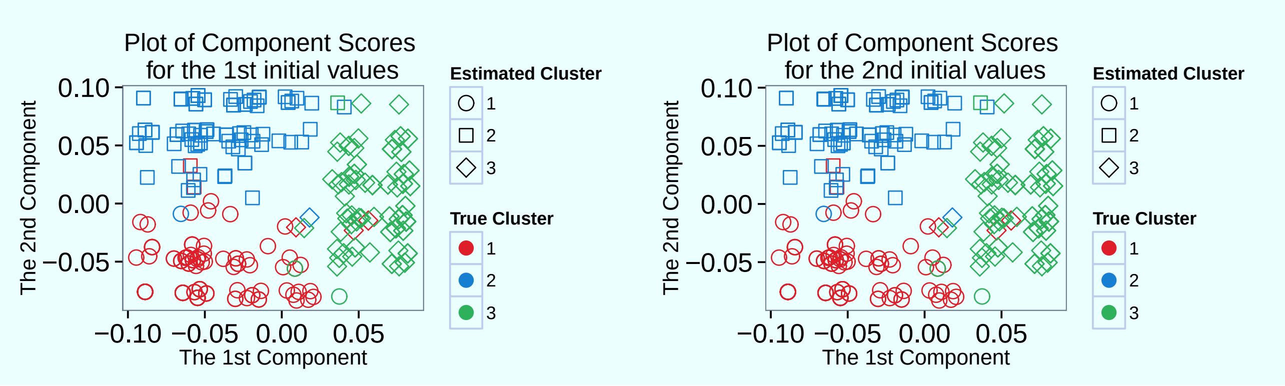


Figure 3: Plots of component scores estimated by CLUSBIRD using the two initial starts that provide the same value of the penalized log likelihood for the data set consisting of 10 variables; the colors and shapes of plotted points denote the true and estimated memberships, respectively.

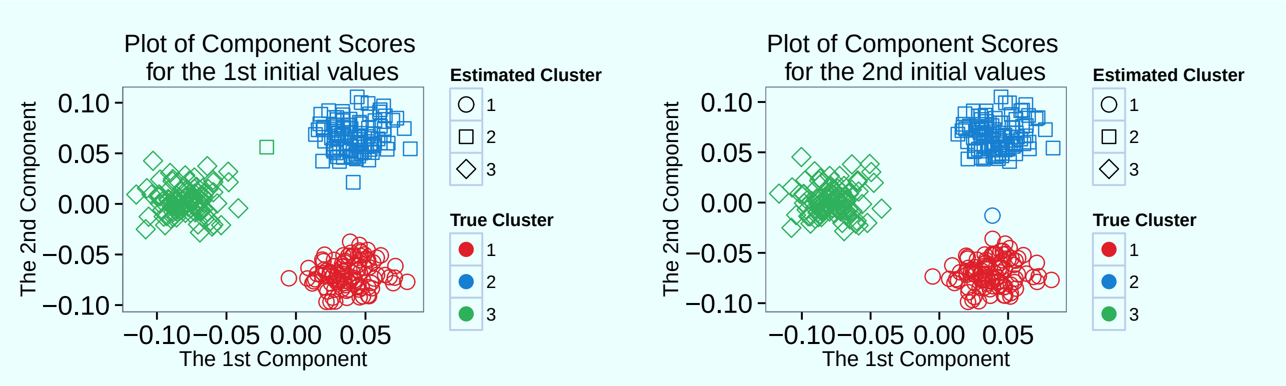


Figure 4: Plots of component scores estimated by CLUSBIRD using the two initial starts that provide the same value of the penalized log likelihood for the data set consisting of 1000 variables.

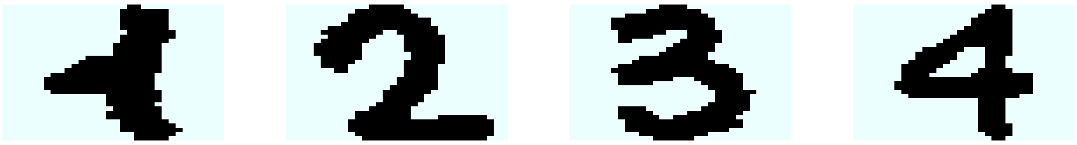
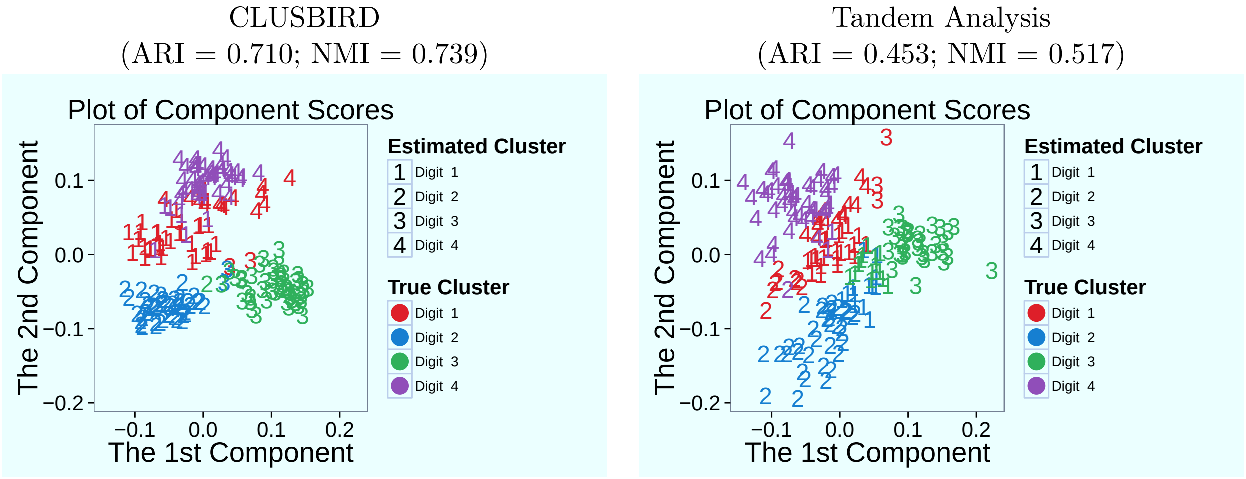


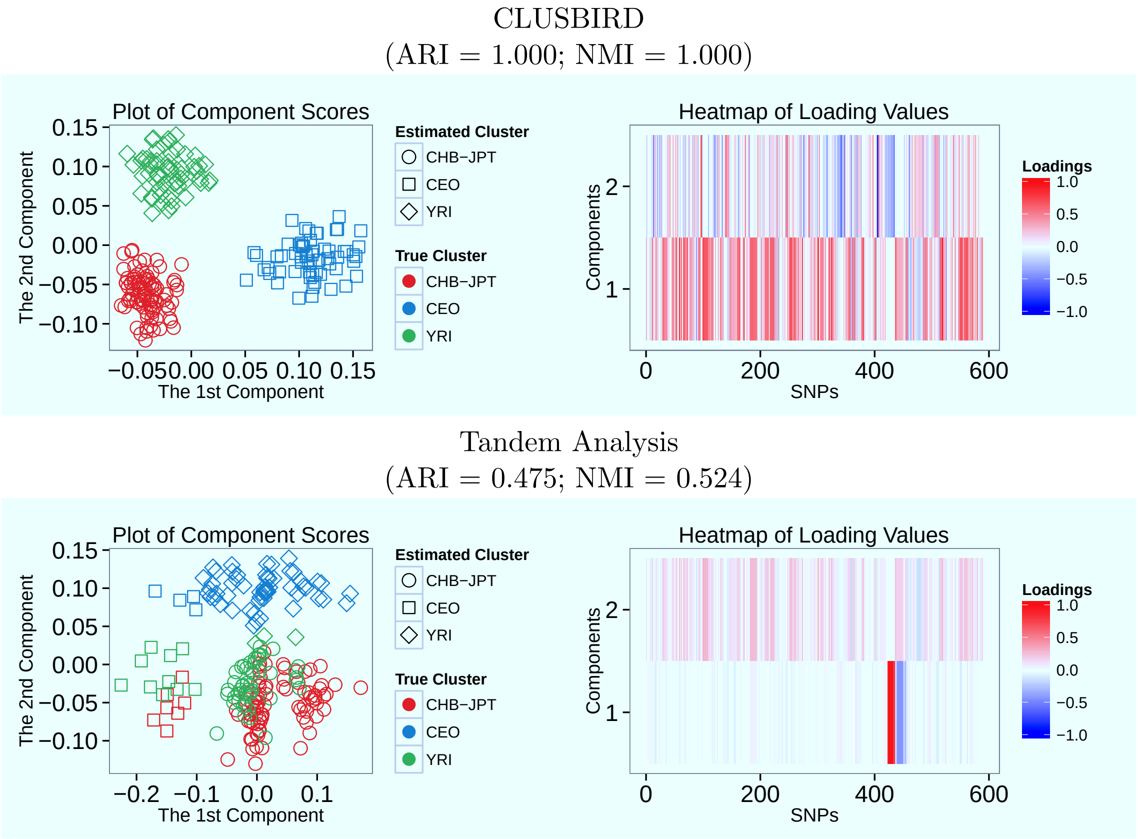
Figure 5: Examples of normalized bitmaps.



CLUSBIRD
(ARI = 0.710; NMI = 0.739)

Tandem Analysis
(ARI = 0.453; NMI = 0.517)

Figure 6: Plots of component scores estimated by CLUSBIRD and tandem analysis; in the plots, the number denotes the estimated cluster and the color denotes the true cluster.



CLUSBIRD
(ARI = 1.000; NMI = 1.000)

Tandem Analysis
(ARI = 0.475; NMI = 0.524)

Figure 7: Plots of component scores (left) and loading values (right) estimated by CLUSBIRD and tandem analysis; in the left panel, the colors and shapes denote the true and estimated memberships, respectively; loading values were scaled so that the value existed in $[-1, 1]$.