



HAL
open science

Parameter-Wise Co-Clustering for High-Dimensional Data

Michael P B Gallagher, Christophe Biernacki, Paul D McNicholas

► **To cite this version:**

Michael P B Gallagher, Christophe Biernacki, Paul D McNicholas. Parameter-Wise Co-Clustering for High-Dimensional Data. 2020. hal-01862824v3

HAL Id: hal-01862824

<https://hal.science/hal-01862824v3>

Preprint submitted on 30 Sep 2020 (v3), last revised 21 Nov 2022 (v4)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Parameter-Wise Co-Clustering for High-Dimensional Data

M.P.B. Gallaugher*, C. Biernacki**, P.D. McNicholas*

*Department of Mathematics & Statistics, McMaster University, Ontario, Canada.

**Laboratory of Mathematics, University Lille 1, Lille, France.

Abstract

In recent years, data dimensionality has increasingly become a concern, leading to many parameter and dimension reduction techniques being proposed in the literature. A parameter-wise co-clustering model, for data modelled via continuous random variables, is presented. The proposed model, although allowing more flexibility, still maintains the very high degree of parsimony achieved by traditional co-clustering. A stochastic expectation-maximization (SEM) algorithm along with a Gibbs sampler is used for parameter estimation and an integrated complete log-likelihood criterion is used for model selection. Simulated and real datasets are used for illustration and comparison with traditional co-clustering.

1 Introduction

Clustering is the process of finding and analyzing underlying group structure in heterogenous data. With the emergence of big data, the number of variables in a dataset is constantly increasing and in many areas of application it is not uncommon for the number of variables to exceed the number of observations. In such situations, where the dimension of the data is very high, traditional mixture modelling techniques for clustering oftentimes fail. Co-clustering is a very useful method for dealing with such scenarios.

Co-clustering aims to define a partition in the rows of the data matrix for clustering individuals, as well as a partition in the columns for clustering variables. The result is partitioning the data matrix into homogenous blocks, or co-clusters, based on both individuals and variables. A key assumption for maintaining parsimony is that observations within each block are realizations of independent and identically distributed random variables. Some of

the earliest work in co-clustering can be traced to Hartigan (1972). Since that time, model-based approaches have recently been shown to be effective for data treated as realizations of a continuous random variable (Nadif & Govaert, 2010), count data (Pledger & Arnold, 2014) and ordinal data (Jacques & Biernacki, 2018), to name but a few. In traditional co-clustering, added flexibility is often obtained by fitting more row-clusters and column-clusters; however, this is not generally advisable for parsimony reasons. Herein, we propose a parameter-wise co-clustering model that separately clusters columns according to both means and variances using the Gaussian distribution.

The remainder of this paper is laid out as follows. Section 2 presents a detailed background on high dimensional clustering techniques as well as details on traditional co-clustering using the Gaussian distribution. Section 3 presents the new model, parameter estimation, model selection criterion, and a non-exhaustive search algorithm for model selection. In Sections 4 and 5, synthetic and real datasets are considered for algorithm evaluation, classification performance, model selection performance, and comparison with traditional co-clustering. We conclude with a discussion of the results (Section 6).

2 Gaussian-Based Clustering for High Dimensional Data

2.1 Model-Based Clustering

Consider a dataset $\mathbf{x} = (\mathbf{x}'_1, \mathbf{x}'_2, \dots, \mathbf{x}'_n)'$ with n individuals $\mathbf{x}_i \in \mathbb{R}^p$. One common method for clustering is model-based clustering, and generally makes use of a finite mixture model. A finite mixture model assumes that a real random vector \mathbf{X}_i of dimension p has probability density function

$$f(\mathbf{x}_i|\boldsymbol{\vartheta}) = \sum_{g=1}^G \pi_g f(\mathbf{x}_i|\boldsymbol{\Theta}_g),$$

where $\pi_g > 0 \forall g$ and $\sum_{g=1}^G \pi_g = 1$ are the mixing proportions, $f(\cdot|\boldsymbol{\Theta}_g)$ are the component density functions parameterized by $\boldsymbol{\Theta}_g$, and $\boldsymbol{\vartheta} = (\pi_1, \dots, \pi_G, \boldsymbol{\Theta}_1, \dots, \boldsymbol{\Theta}_G)$ represents all the mixture parameters.

Because of its mathematical tractability, the multivariate Gaussian mixture model is widely studied in the literature. In this case, each of the component densities is a multivariate Gaussian with density

$$f(\mathbf{x}_i|\boldsymbol{\Theta}_g) = \frac{1}{(2\pi)^{\frac{p}{2}}|\boldsymbol{\Sigma}_g|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_g)'\boldsymbol{\Sigma}_g^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_g)\right\},$$

where $\boldsymbol{\Theta}_g = (\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$. The number of free parameters in a Gaussian mixture model is

$$\#\text{Params}_{\text{GaussMix}} = (G - 1) + Gp + Gp(p + 1)/2. \quad (1)$$

Clearly, the number of free parameters in (1) is quadratic in the dimension of the data. As a result, using this simple mixture of Gaussian distributions will usually fail when the dimension p increases.

In traditional model-based clustering, the group membership for observation \mathbf{x}_i is usually represented by the vector $\mathbf{z}_i = (z_{i1}, z_{i2}, \dots, z_{iG})$, where $z_{ig} = 1$ if observation \mathbf{x}_i belongs to group g and 0 otherwise. Moreover, \mathbf{z}_i is a realization of $\mathbf{Z}_i \sim \text{multinomial}(1; \boldsymbol{\pi})$ where $\boldsymbol{\pi} = (\pi_1, \pi_2, \dots, \pi_G)$. In addition, all couples $(\mathbf{X}_i, \mathbf{Z}_i)$ are usually assumed to be independent.

The use of a Gaussian mixture model for clustering can be traced back to Wolfe (1965). Other early work on Gaussian mixture models can be found in Baum et al. (1970) and Scott & Symons (1971). A detailed review of model-based clustering and classification is given by McNicholas (2016), including related estimation and model selection procedures.

2.2 High Dimensional Clustering Techniques

Although the Gaussian mixture model is widely used, problems arise when the data dimensionality p increases. The main contribution to the number of free parameters is through the component covariance matrices $\boldsymbol{\Sigma}_g$. Therefore, as a starting point, many methods try to impose parsimonious constraints on $\boldsymbol{\Sigma}_g$. A detailed background is presented by Bouveyron & Brunet-Saumard (2014) and McNicholas (2016).

One particular example to note is the mixture of factor analyzers model. This model, presented by Ghahramani & Hinton (1997), is a Gaussian mixture model with covariance structure $\boldsymbol{\Sigma}_g = \boldsymbol{\Lambda}_g \boldsymbol{\Lambda}_g' + \boldsymbol{\Psi}$, where $\boldsymbol{\Lambda}_g$ is a $p \times q$ matrix of factor loadings with $q < p$ and $\boldsymbol{\Psi} = \text{diag}(\psi_1, \psi_2, \dots, \psi_p)$, $\psi_j \in \mathbb{R}^+$. Numerous extensions are proposed in the literature, including McLachlan & Peel (2000), who utilize the more general structure $\boldsymbol{\Sigma}_g = \boldsymbol{\Lambda}_g \boldsymbol{\Lambda}_g' + \boldsymbol{\Psi}_g$, and the closely-related mixture of probabilistic principal component analyzers with $\boldsymbol{\Sigma}_g = \boldsymbol{\Lambda}_g \boldsymbol{\Lambda}_g' + \psi_g \mathbf{I}$ (Tipping & Bishop, 1999). In addition to these minor extensions, McNicholas & Murphy (2008) construct a family of eight parsimonious Gaussian models by considering the constraint $\boldsymbol{\Lambda}_g = \boldsymbol{\Lambda}$ in addition to $\boldsymbol{\Psi}_g = \boldsymbol{\Psi}$ and $\boldsymbol{\Psi}_g = \psi_g \mathbf{I}$. For the fully constrained model in McNicholas & Murphy (2008), there are

$$\#\text{Params}_{\text{MFA}} = (G - 1) + Gp + pq - q(q - 1)/2 + 1 \quad (2)$$

free parameters. It is clear that although the number of free parameters associated with these models is linear in p , it is still nevertheless dependent on the dimension. Consequently, these models are still not suitable for very high dimensional data. Moreover, these methods may not be viable when $n > p$, which is common in applications such as gene expression data, word processing data, single nucleotide polymorphism data, etc.

Alternatively, Bouveyron et al. (2007) use the spectral decomposition of Σ_g , i.e., $\Sigma_g = \mathbf{D}_g \mathbf{\Delta}_g \mathbf{D}_g'$, where \mathbf{D}_g is the orthogonal matrix of eigenvectors and $\mathbf{\Delta}_g$ is a diagonal matrix of corresponding eigenvalues for which they impose the structure

$$\mathbf{\Delta}_g = \text{diag}(a_{1g}, a_{2g}, \dots, a_{q_g g}, b_g, b_g, \dots, b_g),$$

where a_{kg} are the q_g largest eigenvalues and b_g is average of the remaining $p - q_g$ eigenvalues. This also greatly reduces the number of free parameters, i.e.,

$$\#\text{Params}_{\text{Bouveyron}} = (G - 1) + Gp + \sum_{g=1}^G q_g [p - (q_g + 1)/2] + \sum_{g=1}^G q_g + 2G. \quad (3)$$

Again, however, the number of free parameters is dependent on the dimensionality of the data.

Finally, there are also variable selection procedures such as ℓ_1 penalization methods which take advantage of sparsity to perform variable selection and parameter estimation simultaneously. The first such proposed method is presented by Pan & Shen (2007) who consider equal, diagonal covariance matrices between groups and apply an ℓ_1 penalty to the mean vectors. A lasso method is then used for parameter estimation. This is extended by Zhou et al. (2009), who consider unconstrained covariance matrices and apply an ℓ_1 penalty for both the mean and covariance parameters. Although these methods are useful for dealing with the dimensionality problem, the ℓ_1 penalty shrinks the parameters, thus introducing bias, as discussed by Meynet & Maugis-Rabuseau (2012). Moreover, the Bayesian information criterion (BIC; Schwarz, 1978) may not be suitable for high-dimensional data. A detailed review of each of these methods is given by Biernacki & Maugis (2017).

2.3 Co-Clustering and its Limitations

Co-Clustering is a very useful tool for analyzing high-dimensional data. This method considers simultaneous partitions of rows and columns, which are then used to organize the data into homogenous blocks. For traditional co-clustering, as in clustering, data are assumed to come in the form of an $n \times p$ matrix \mathbf{x} with rows represented by \mathbf{x}_i' . Each individual element of \mathbf{x}_i is denoted by x_{ij} , so that x_{ij} is the observation in row i and column j .

In co-clustering, there is an unknown partition of the rows into G clusters, from this point onwards referred to as row-clusters, represented by the indicator vector \mathbf{z}_i as defined previously. Unlike traditional co-clustering, however, there is also a partition of the columns into L clusters, referred to as column-clusters, represented by the indicator vector $\mathbf{w}_j = (w_{j1}, w_{j2}, \dots, w_{jL}) \sim \text{multinomial}(1; \boldsymbol{\rho})$, where $w_{jl} = 1$ if column j belongs to column-cluster l and $w_{jl} = 0$ otherwise, and $\boldsymbol{\rho} = (\rho_1, \rho_2, \dots, \rho_L)$. It is assumed that each data point

x_{ij} is independent once the \mathbf{z}_i and \mathbf{w}_j are fixed. If, in addition, all \mathbf{z}_i and \mathbf{w}_j are assumed independent, and the latent block model is utilized in the same manner as Nadif & Govaert (2010), then the joint density of \mathbf{x} becomes $f(\mathbf{x}; \boldsymbol{\vartheta}) = \sum_{\mathbf{z} \in \mathcal{Z}} \sum_{\mathbf{w} \in \mathcal{W}} p(\mathbf{z}; \boldsymbol{\pi}) p(\mathbf{w}; \boldsymbol{\rho}) f(\mathbf{x} | \mathbf{z}, \mathbf{w}; \boldsymbol{\Theta})$, where

$$p(\mathbf{z}; \boldsymbol{\pi}) = \prod_{i=1}^n \prod_{g=1}^G \pi_g^{z_{ig}}, \quad p(\mathbf{w}; \boldsymbol{\rho}) = \prod_{j=1}^p \prod_{l=1}^L \rho_l^{w_{jl}}, \quad \text{and}$$

$$f(\mathbf{x} | \mathbf{z}, \mathbf{w}^\mu, \mathbf{w}^\Sigma; \boldsymbol{\Theta}) = \prod_{i=1}^n \prod_{g=1}^G \prod_{j=1}^p \prod_{l=1}^L \left[\frac{1}{\sqrt{2\pi}\sigma_{gl}} \exp \left\{ -\frac{1}{2\sigma_{gl}^2} (x_{ij} - \mu_{gl})^2 \right\} \right]^{z_{ig} w_{jl}},$$

where μ_{gl} and σ_{gl}^2 are the mean and variance, respectively, for row-cluster g and column-cluster l , $\boldsymbol{\Theta}$ is the set of all μ_{gl} and σ_{gl}^2 , and $\boldsymbol{\vartheta} = (\boldsymbol{\pi}, \boldsymbol{\rho}, \boldsymbol{\Theta})$. The total number of free parameters in this traditional co-clustering model is

$$\#\text{Params}_{\text{trad coclust}} = G + L + 2(GL - 1). \quad (4)$$

Note that (4) does not depend on the dimension, making it a very parsimonious model. Moreover, co-clustering is still possible to perform when $p > n$.

There are two different ways that one can view co-clustering. The first is that the main goal is the clustering of rows, and the clustering of columns is solely a way to solve the problem of dimensionality. However, in certain applications, the clustering of the columns might also be of interest.

Limitations of Co-Clustering Although co-clustering has advantages over other high dimensional techniques (especially in the number of free parameters), the model is fairly restrictive because all observations in a block are realizations of independent and identically distributed Gaussian random variables with mean μ_{gl} and variance σ_{gl}^2 . More flexibility is obtained by fitting more column-clusters and row-clusters, which is not always possible or advisable. What we propose in the present work is a parameter-wise co-clustering method by clustering columns according to both means and variances. This is the reason why we adopt hereafter the denomination ‘‘parameter-wise’’ co-clustering, which is now presented in detail.

3 Parameter-Wise Gaussian Co-Clustering

3.1 A Model to Combine Two Latent Variables in Columns

Recall that traditional co-clustering aims to cluster data such that observations in the same block have the same distribution. An extension of traditional co-clustering for data treated

as realizations of a Gaussian random variable is now considered. Similar to traditional clustering, there is a partition in rows and columns. However, now there are two partitions in the columns; specifically, a partition with respect to means and a partition with respect to variances.

Recall also that the data, which are treated as realizations of a continuous random variable, are represented as an $n \times p$ matrix, $\mathbf{x} = (x_{ij})_{1 \leq i \leq n, 1 \leq j \leq p}$. The partition in rows is again represented by $\mathbf{z} = (\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n)$.

Two Partitions in Columns The partition in columns by means is represented by $\mathbf{w}^\mu = (\mathbf{w}_1^\mu, \mathbf{w}_2^\mu, \dots, \mathbf{w}_p^\mu)$, where

$$\mathbf{w}_j^\mu = (w_{j1}^\mu, w_{j2}^\mu, \dots, w_{jL^\mu}^\mu) \sim \text{multinomial}(1; \boldsymbol{\rho}^\mu)$$

with $\boldsymbol{\rho}^\mu = (\rho_1^\mu, \rho_2^\mu, \dots, \rho_{L^\mu}^\mu)$ and the partition in columns by variances is denoted by $\mathbf{w}^\Sigma = (\mathbf{w}_1^\Sigma, \mathbf{w}_2^\Sigma, \dots, \mathbf{w}_p^\Sigma)$, where

$$\mathbf{w}_j^\Sigma = (w_{j1}^\Sigma, w_{j2}^\Sigma, \dots, w_{jL^\Sigma}^\Sigma) \sim \text{multinomial}(1; \boldsymbol{\rho}^\Sigma)$$

with $\boldsymbol{\rho}^\Sigma = (\rho_1^\Sigma, \rho_2^\Sigma, \dots, \rho_{L^\Sigma}^\Sigma)$. These two partitions in the columns is where the main novelty lies. Note that G, L^μ and L^Σ are the number of row-clusters, column-clusters by means, and column-clusters by variances, respectively.

Log-Likelihood Using a small extension of the latent block model the observed log-likelihood is then

$$f(\mathbf{x}; \boldsymbol{\theta}) = \sum_{\mathbf{z} \in \mathcal{Z}} \sum_{\mathbf{w}^\mu \in \mathcal{W}^\mu} \sum_{\mathbf{w}^\Sigma \in \mathcal{W}^\Sigma} p(\mathbf{z}; \boldsymbol{\pi}) p(\mathbf{w}^\mu; \boldsymbol{\rho}^\mu) p(\mathbf{w}^\Sigma; \boldsymbol{\rho}^\Sigma) f(\mathbf{x} | \mathbf{z}, \mathbf{w}^\mu, \mathbf{w}^\Sigma; \boldsymbol{\mu}, \boldsymbol{\Sigma}),$$

where

$$p(\mathbf{z}; \boldsymbol{\pi}) = \prod_{i=1}^n \prod_{g=1}^G \pi_g^{z_{ig}}, \quad p(\mathbf{w}^\mu; \boldsymbol{\rho}^\mu) = \prod_{j=1}^p \prod_{l^\mu=1}^{L^\mu} (\rho_{l^\mu}^\mu)^{w_{jl^\mu}^\mu}, \quad p(\mathbf{w}^\Sigma; \boldsymbol{\rho}^\Sigma) = \prod_{j=1}^p \prod_{l^\Sigma=1}^{L^\Sigma} (\rho_{l^\Sigma}^\Sigma)^{w_{jl^\Sigma}^\Sigma}, \quad \text{and}$$

$$f(\mathbf{x} | \mathbf{z}, \mathbf{w}^\mu, \mathbf{w}^\Sigma; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \prod_{i=1}^n \prod_{g=1}^G \prod_{j=1}^p \prod_{l^\mu=1}^{L^\mu} \prod_{l^\Sigma=1}^{L^\Sigma} \left[\frac{1}{\sqrt{2\pi}\sigma_{gl^\Sigma}} \exp \left\{ -\frac{1}{2\sigma_{gl^\Sigma}^2} (x_{ij} - \mu_{gl^\mu})^2 \right\} \right]^{z_{ig} w_{jl^\mu}^\mu w_{jl^\Sigma}^\Sigma}.$$

In terms of notation, $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_G)$, where $\boldsymbol{\mu}_g = (\mu_{g1}, \mu_{g2}, \dots, \mu_{gL^\mu})$. Note that μ_{gl^μ} is the mean for row-cluster g and column-cluster by means l^μ . Likewise, $\boldsymbol{\Sigma} = (\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2, \dots, \boldsymbol{\Sigma}_G)$, where $\boldsymbol{\Sigma}_g = (\sigma_{g1}^2, \sigma_{g2}^2, \dots, \sigma_{gL^\Sigma}^2)$ and $\sigma_{gl^\Sigma}^2$ is the variance for row-cluster g and column-cluster

by variances l^Σ . Finally, the complete-data log-likelihood is

$$p(\mathbf{x}, \mathbf{z}, \mathbf{w}^\mu, \mathbf{w}^\Sigma; \boldsymbol{\vartheta}) = C + \sum_{i=1}^n \sum_{g=1}^G z_{ig} \log \pi_g + \sum_{j=1}^p \sum_{l^\mu=1}^{L^\mu} w_{jl^\mu}^\mu \log \rho_{l^\mu}^\mu + \sum_{j=1}^p \sum_{l^\Sigma=1}^{L^\Sigma} w_{jl^\Sigma}^\Sigma \log \rho_{l^\Sigma}^\Sigma - \frac{1}{2} \sum_{i=1}^n \sum_{g=1}^G \sum_{j=1}^p \sum_{l^\mu=1}^{L^\mu} \sum_{l^\Sigma=1}^{L^\Sigma} z_{ig} w_{jl^\mu}^\mu w_{jl^\Sigma}^\Sigma \left[\log \sigma_{gl^\Sigma}^2 + \frac{(x_{ij} - \mu_{gl^\mu})^2}{\sigma_{gl^\Sigma}^2} \right],$$

where C is a constant with respect to the parameters and $\boldsymbol{\vartheta} = (\boldsymbol{\pi}, \boldsymbol{\rho}^\mu, \boldsymbol{\rho}^\Sigma, \boldsymbol{\mu}, \boldsymbol{\Sigma})$. From this point on, we refer to this model as parameter-wise co-clustering.

Number of Free Parameters The number of free parameters in the parameter-wise co-clustering model is

$$\begin{aligned} \#\text{Params}_{\text{new coclust}} &= G - 1 + L^\mu - 1 + L^\Sigma - 1 + GL^\mu + GL^\Sigma \\ &= G + (L^\mu + L^\Sigma)(G + 1) - 3. \end{aligned}$$

There are a few comparisons with traditional co-clustering that are now discussed. First, similar to traditional co-clustering, the number of free parameters for the proposed parameter-wise method is independent of the dimension, meaning a high degree of parsimony is still maintained. Before mentioning the second point, note that the column-clusters by means and column-clusters by variances can be combined. For example, columns in column-cluster 1 by means and column-cluster 1 by variances can be combined to form one column-cluster. In general, columns in column-cluster l^μ by means and column-cluster l^Σ by variances can be combined to form one column-cluster for any combination of l^μ and l^Σ , leading to a maximum of $L^\mu L^\Sigma$ column-clusters. There can, however, be fewer than $L^\mu L^\Sigma$ combined column-clusters because it is possible, for example, that no columns are clustered into column-cluster 3 by means and column-cluster 2 by variances. Now, assuming G is equal for both parameter-wise and traditional co-clustering, and $L^\mu = L^\Sigma = L$, then there are only an additional $L - 1$ free parameters when using the parameter-wise model. Although there are these additional free parameters, there is the possibility of L^2 combined column-clusters, allowing for a finer partition of the columns and increased flexibility.

There is also the possibility that the parameter-wise model has fewer free parameters than traditional co-clustering while still maintaining similar flexibility. For example, if traditional co-clustering is considered with $G = 4$ and $L = 5$, then the total number of free parameters is 47. In the parameter-wise case, if $G = 4$, $L^\mu = 3$, $L^\Sigma = 3$, then the total number of free parameters is 31. In this case, there is a possibility of a total of nine column-clusters compared to five column-clusters when using traditional co-clustering.

3.2 Parameter Estimation Using the SEM Gibbs Algorithm

The SEM algorithm after initialization at iteration q proceeds as follows.

SE Step: Generate the row partition $\mathbf{z}^{(q+1)}$ according to

$$P(z_{ig} = 1 | \mathbf{x}, \mathbf{w}^{\mu^{(q)}}, \mathbf{w}^{\Sigma^{(q)}}; \boldsymbol{\mu}^{(q)}, \boldsymbol{\Sigma}^{(q)}, \boldsymbol{\pi}^{(q)}) = \frac{\pi_g^{(q)} f(\mathbf{x}_i | \mathbf{w}^{\mu^{(q)}}, \mathbf{w}^{\Sigma^{(q)}}; \boldsymbol{\mu}_g^{(q)}, \boldsymbol{\Sigma}_g^{(q)})}{\sum_{g'}^G \pi_{g'}^{(q)} f(\mathbf{x}_i | \mathbf{w}^{\mu^{(q)}}, \mathbf{w}^{\Sigma^{(q)}}; \boldsymbol{\mu}_{g'}^{(q)}, \boldsymbol{\Sigma}_{g'}^{(q)})},$$

where

$$f(\mathbf{x}_i | \mathbf{w}^{\mu^{(q)}}, \mathbf{w}^{\Sigma^{(q)}}; \boldsymbol{\mu}_g^{(q)}, \boldsymbol{\Sigma}_g^{(q)}) = \prod_{j=1}^p \prod_{l^\mu=1}^{L^\mu} \prod_{l^\Sigma=1}^{L^\Sigma} \left[\frac{1}{\sqrt{2\pi}\sigma_{gl^\Sigma}^{(q)}} \exp \left\{ -\frac{1}{2\sigma_{gl^\Sigma}^{2(q)}} (x_{ij} - \mu_{gl^\mu}^{(q)})^2 \right\} \right]^{w_{jl^\mu}^{\mu^{(q)}} w_{jl^\Sigma}^{\Sigma^{(q)}}}.$$

Generate the column partition by means $\mathbf{w}^{\mu^{(q+1)}}$ according to

$$P(w_{jl^\mu}^\mu = 1 | \mathbf{x}, \mathbf{z}^{(q+1)}, \mathbf{w}^{\Sigma^{(q)}}; \boldsymbol{\mu}^{(q)}, \boldsymbol{\Sigma}^{(q)}, \boldsymbol{\rho}^{\mu^{(q)}}) = \frac{\rho_{l^\mu}^{\mu^{(q)}} f(\mathbf{x}_{\cdot j} | \mathbf{z}^{(q+1)}, \mathbf{w}^{\Sigma^{(q)}}; \boldsymbol{\mu}_{l^\mu}^{(q)}, \boldsymbol{\Sigma}_{l^\mu}^{(q)})}{\sum_{l^{\mu'}}^{L^\mu} \rho_{l^{\mu'}}^{\mu^{(q)}} f(\mathbf{x}_{\cdot j} | \mathbf{z}^{(q+1)}, \mathbf{w}^{\Sigma^{(q)}}; \boldsymbol{\mu}_{l^{\mu'}}^{(q)}, \boldsymbol{\Sigma}_{l^{\mu'}}^{(q)})},$$

where $\mathbf{x}_{\cdot j} = (x_{1j}, x_{2j}, \dots, x_{nj})$, $\boldsymbol{\mu}_{l^\mu}^{(q)} = (\mu_{1l^\mu}^{(q)}, \mu_{2l^\mu}^{(q)}, \dots, \mu_{Gl^\mu}^{(q)})$, and

$$f(\mathbf{x}_{\cdot j} | \mathbf{z}^{(q+1)}, \mathbf{w}^{\Sigma^{(q)}}; \boldsymbol{\mu}_{l^\mu}^{(q)}, \boldsymbol{\Sigma}_{l^\mu}^{(q)}) = \prod_{i=1}^n \prod_{g=1}^G \prod_{l^\Sigma=1}^{L^\Sigma} \left[\frac{1}{\sqrt{2\pi}\sigma_{gl^\Sigma}^{(q)}} \exp \left\{ -\frac{1}{2\sigma_{gl^\Sigma}^{2(q)}} (x_{ij} - \mu_{gl^\mu}^{(q)})^2 \right\} \right]^{z_{ig}^{(q+1)} w_{jl^\mu}^{\mu^{(q)}} w_{jl^\Sigma}^{\Sigma^{(q)}}}.$$

Generate the column partition by variances $\mathbf{w}^{\Sigma^{(q+1)}}$ according to

$$P(w_{jl^\Sigma}^\Sigma = 1 | \mathbf{x}, \mathbf{z}^{(q+1)}, \mathbf{w}^{\mu^{(q+1)}}; \boldsymbol{\mu}^{(q)}, \boldsymbol{\Sigma}^{(q)}, \boldsymbol{\rho}^{\Sigma^{(q)}}) = \frac{\rho_{l^\Sigma}^{\Sigma^{(q)}} f(\mathbf{x}_{\cdot j} | \mathbf{z}^{(q+1)}, \mathbf{w}^{\mu^{(q+1)}}; \boldsymbol{\mu}_{l^\Sigma}^{(q)}, \boldsymbol{\Sigma}_{l^\Sigma}^{(q)})}{\sum_{l^{\Sigma'}}^{L^\Sigma} \rho_{l^{\Sigma'}}^{\Sigma^{(q)}} f(\mathbf{x}_{\cdot j} | \mathbf{z}^{(q+1)}, \mathbf{w}^{\mu^{(q+1)}}; \boldsymbol{\mu}_{l^{\Sigma'}}^{(q)}, \boldsymbol{\Sigma}_{l^{\Sigma'}}^{(q)})},$$

where $\boldsymbol{\Sigma}_{l^\Sigma}^{(q)} = (\sigma_{1l^\Sigma}^{2(q)}, \sigma_{2l^\Sigma}^{2(q)}, \dots, \sigma_{Gl^\Sigma}^{2(q)})$ and

$$f(\mathbf{x}_{\cdot j} | \mathbf{z}^{(q+1)}, \mathbf{w}^{\mu^{(q+1)}}; \boldsymbol{\mu}_{l^\Sigma}^{(q)}, \boldsymbol{\Sigma}_{l^\Sigma}^{(q)}) = \prod_{i=1}^n \prod_{g=1}^G \prod_{l^\mu=1}^{L^\mu} \left[\frac{1}{\sqrt{2\pi}\sigma_{gl^\Sigma}^{(q)}} \exp \left\{ -\frac{1}{2\sigma_{gl^\Sigma}^{2(q)}} (x_{ij} - \mu_{gl^\mu}^{(q)})^2 \right\} \right]^{z_{ig}^{(q+1)} w_{jl^\mu}^{\mu^{(q+1)}}}.$$

M Step: Update the parameters according to

$$\begin{aligned} \pi_g^{(q+1)} &= \frac{\sum_{i=1}^n z_{ig}^{(q+1)}}{n}, & \rho_{l^\mu}^{\mu^{(q+1)}} &= \frac{\sum_{j=1}^p w_{jl^\mu}^{\mu^{(q+1)}}}{p}, & \rho_{l^\Sigma}^{\Sigma^{(q+1)}} &= \frac{\sum_{j=1}^p w_{jl^\Sigma}^{\Sigma^{(q+1)}}}{p}, \\ \mu_{gl^\mu}^{(q+1)} &= \frac{\sum_{i=1}^n \sum_{j=1}^p \sum_{l^\Sigma=1}^{L^\Sigma} z_{ig}^{(q+1)} w_{jl^\mu}^{\mu^{(q+1)}} w_{jl^\Sigma}^{\Sigma^{(q+1)}} x_{ij}}{\sum_{i=1}^n \sum_{j=1}^p \sum_{l^\Sigma=1}^{L^\Sigma} z_{ig}^{(q+1)} w_{jl^\mu}^{\mu^{(q+1)}} w_{jl^\Sigma}^{\Sigma^{(q+1)}}} = \frac{\sum_{i=1}^n \sum_{j=1}^p z_{ig}^{(q+1)} w_{jl^\mu}^{\mu^{(q+1)}} x_{ij}}{\sum_{i=1}^n \sum_{j=1}^p z_{ig}^{(q+1)} w_{jl^\mu}^{\mu^{(q+1)}}}, \\ \sigma_{gl^\Sigma}^{2(q+1)} &= \frac{\sum_{i=1}^n \sum_{j=1}^p \sum_{l^\mu=1}^{L^\mu} z_{ig}^{(q+1)} w_{jl^\mu}^{\mu^{(q+1)}} w_{jl^\Sigma}^{\Sigma^{(q+1)}} (x_{ij} - \mu_{gl^\mu}^{(q+1)})^2}{\sum_{i=1}^n \sum_{j=1}^p \sum_{l^\mu=1}^{L^\mu} z_{ig}^{(q+1)} w_{jl^\mu}^{\mu^{(q+1)}} w_{jl^\Sigma}^{\Sigma^{(q+1)}}}. \end{aligned}$$

After a burn-in period of the algorithm, the estimates of each of the parameters are just the mean of the runs of the SEM algorithm (the number of runs are assessed experimentally in Section 4). We denote these final estimates by $\hat{\boldsymbol{\vartheta}} = (\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\rho}}^\mu, \hat{\boldsymbol{\rho}}^\Sigma, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$. For the final partition of rows, columns by means, and columns by variances, we fix the parameters at their estimates and run more iterations of the SE step. We then assign each row to the row-cluster to which it is assigned most often over these additional SE steps. Likewise, each column is assigned to the column-cluster by means to which it is assigned most often over the additional SE steps, and finally each column is assigned to the column-cluster by variances to which it is assigned most often over the additional SE iterations. For our simulations and real data analyses, we take 20 such runs to obtain the final partitions $\hat{\mathbf{z}}$, $\hat{\mathbf{w}}^\mu$, and $\hat{\mathbf{w}}^\Sigma$.

3.3 Model Selection

ICL–BIC As is the case in any clustering scenario, the number of row-clusters, column-clusters by means, and column-clusters by variances are not known *a priori* and, therefore, a model selection criterion is required. Similar to traditional co-clustering, the observed log-likelihood is intractable and so the BIC cannot be used. Therefore, we propose using the integrated complete log-likelihood (ICL; Biernacki et al., 2000), which relies on the complete data log-likelihood instead of the observed log-likelihood. This criterion is called the ICL–BIC, similar to that used by Jacques & Biernacki (2018) and is given by

$$\text{ICL–BIC} = p(\mathbf{x}, \hat{\mathbf{z}}, \hat{\mathbf{w}}^\mu, \hat{\mathbf{w}}^\Sigma; \hat{\boldsymbol{\vartheta}}) - \frac{G-1}{2} \log n - \frac{L^\mu + L^\Sigma - 2}{2} \log p - \frac{G(L^\mu + L^\Sigma)}{2} \log np.$$

From the property proven by Brault et al. (2017), the BIC and ICL–BIC exhibit the same behaviour for large values of n and/or p , thus the number of blocks chosen by this criterion is consistent (under some conditions not mentioned here). The model with the largest ICL–BIC is retained.

Search Algorithm Because an extra layer of complexity is introduced with the parameter-wise model by considering two column partitions, it may take a very long time to perform an exhaustive search of all possible combinations of G , L^μ and L^Σ in a pre-defined range. This has been discussed in the literature, specifically by Robert (2017), and a non-exhaustive search algorithm for the parameter-wise model is now presented. Specifically, the algorithm begins with the parameters $(G, L^\mu, L^\Sigma) = (G_1, L_1^\mu, L_1^\Sigma)$. Three models with parameters $(G_1 + 1, L^\mu, L^\Sigma)$, $(G_1, L^\mu + 1, L^\Sigma)$ and $(G_1, L^\mu, L^\Sigma + 1)$ are then fit. The set with the highest ICL–BIC is retained and we obtain the set $(G_2, L_2^\mu, L_2^\Sigma)$. The procedure is then repeated until a maximum threshold is reached for these parameters or the ICL–BIC no longer in-

creases. Although not as pertinent for traditional co-clustering, a similar non-exhaustive search algorithm can be used for traditional co-clustering.

4 Numerical Experiments on Artificial Data

4.1 Algorithm and Parameter Estimation Evaluation

Two different simulations are performed to evaluate the algorithm, parameter estimation, and classification performance.

Simulation 1

50 datasets are simulated according to the following parameters. $n = 1000$, $p = 100$, $G = 3$, $L^\mu = 2$, $L^\Sigma = 3$,

$$\boldsymbol{\mu} = \begin{pmatrix} 1 & -1 \\ 2 & -2 \\ 3 & -3 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.5 & 0.75 \\ 2 & 1.75 & 0.25 \\ 1.5 & 2.25 & 2.5 \end{pmatrix},$$

and mixing proportions

$$\boldsymbol{\pi} = (0.3, 0.3, 0.4), \quad \boldsymbol{\rho}^\mu = (0.4, 0.6), \quad \boldsymbol{\rho}^\Sigma = (0.3, 0.3, 0.4).$$

To clarify notation, the cell gl^μ in the matrix $\boldsymbol{\mu}$ corresponds to the mean of an observation from row-cluster g and column-cluster by means l^μ , i.e., μ_{gl^μ} . Likewise, the cell gl^Σ in the matrix $\boldsymbol{\Sigma}$ corresponds to the variance of an observation from row-cluster g and column-cluster by variances l^Σ , i.e., $\sigma_{gl^\Sigma}^2$.

A burn-in of 20 iterations for the SEM-Gibbs algorithm is used, followed by 100 iterations, followed by 20 iterations of the SE-step to obtain the final partitions.

The error in the mean estimates is calculated using

$$\Delta\boldsymbol{\mu} = \sum_{g,l^\mu} |\hat{\mu}_{gl^\mu} - \mu_{gl^\mu}|.$$

The errors for the other parameters are calculated in a similar fashion and are denoted by $\Delta\boldsymbol{\Sigma}$, $\Delta\boldsymbol{\pi}$, $\Delta\boldsymbol{\rho}^\mu$ and $\Delta\boldsymbol{\rho}^\Sigma$, respectively. The averaged errors (and their standard deviations) over the 50 datasets are shown in Table 1. The average errors are low for all variables indicating good parameter recovery.

The adjusted rand index (ARI; Hubert & Arabie, 1985) is used to assess classification performance. This quantity compares two partitions, in this case the true partition to an estimated partition, and has a value of 1 if there is perfect agreement, and an expected

value of 0 under random classification. Table 2 displays the average ARI, with standard deviations, for the row, column by means, and column by variances partitions over the 50 simulated datasets. Notice that the classification is perfect for both partitions by columns for all simulated datasets. Moreover, the average ARI for the rows is very high.

Table 1: Average error (and standard deviation) of the parameter estimates over the 50 datasets for Simulation 1.

$\overline{\Delta\mu}$	$\overline{\Delta\Sigma}$	$\overline{\Delta\pi}$	$\overline{\Delta\rho^\mu}$	$\overline{\Delta\rho^\Sigma}$
0.14 (0.70)	0.24 (0.75)	0.012 (0.082)	1.44e-15 (5.61e-16)	1.33e-15 (4.59e-16)

Table 2: Average ARI (and standard deviation) for the row ($\overline{\text{ARI}}_r$), column by means ($\overline{\text{ARI}}_{c\mu}$), and column by variances ($\overline{\text{ARI}}_{c\Sigma}$) partitions over the 50 datasets for Simulation 1.

$\overline{\text{ARI}}_r$	$\overline{\text{ARI}}_{c\mu}$	$\overline{\text{ARI}}_{c\Sigma}$
0.99 (0.068)	1.00 (0.00)	1.00 (0.00)

In Figure 1, the progression of the parameter estimates over the course of the SEM-Gibbs algorithm is shown for one of the datasets (the other datasets exhibit similar behaviour). From these plots, it is clear that a burn-in of 20 iterations is sufficient to obtain a stable chain.

Finally, in Figure 2, the co-clustering results for one of the 50 datasets is displayed. Note, in this case, the estimated co-clustering result is the same as the true co-clustering solution. In the top left panel, a heatmap of the original data is displayed. In the co-clustering by means panel (bottom left), the co-clustering results for the row-clusters and the column-clusters by means is shown. The co-clustering by variances panel (bottom right) shows the co-clustering results for the row-clusters and the column-clusters by variances. Finally, the combined co-clustering (top right) displays the co-clustering solution with all combined column-clusters. Specifically, going from left to right, the first combined column-cluster consists of the columns partitioned into column-cluster 1 for the means and column-cluster 1 for the variances, the second combined column-cluster are the columns clustered into column-cluster 2 for the means and column-cluster 1 for the variances and so on. Combining the column-clusters by means and variances in this manner results in a maximum of $L^\mu L^\Sigma$ combined column-clusters (as is the case here) thus allowing more flexibility. It is important to note, however, that there may be cases, as we will see with the real dataset, when no columns are clustered into a particular pair l^μ and l^Σ , and thus the combined co-clustering result might have fewer than $L^\mu L^\Sigma$ combined column-clusters but never more.

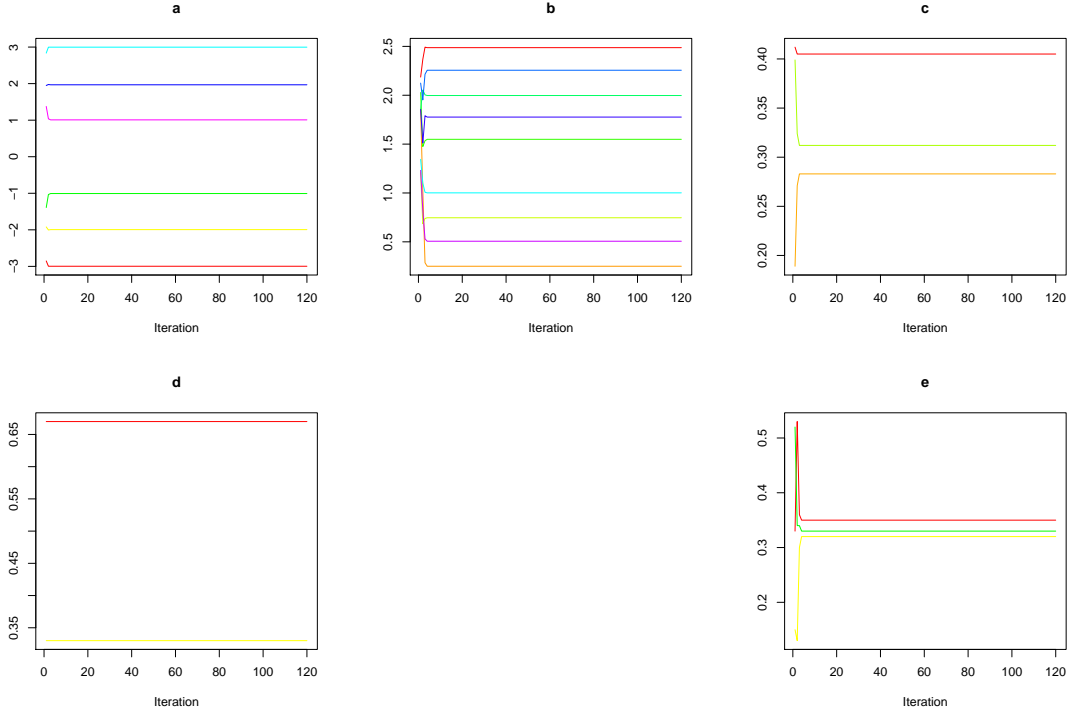


Figure 1: SEM algorithm parameter estimation progression for one dataset for (a) the mean parameters μ_{gl^μ} , (b) the variance parameters $\sigma_{gl^\Sigma}^2$, (c) the row mixing proportions π_g , (d) the column by means mixing proportions $\rho_{l^\mu}^\mu$, and (e) the column by variances mixing proportions $\rho_{l^\Sigma}^\Sigma$ for Simulation 1.

Simulation 2

In Simulation 2, less separation between groups is considered. A total of 50 datasets are again considered with the parameters $n = 200$, $p = 500$, $G = 3$, $L^\mu = 3$, $L^\Sigma = 2$,

$$\boldsymbol{\mu} = \begin{pmatrix} 1 & 1.25 & 0 \\ 2 & 1.2 & 1 \\ 1.5 & 1.9 & 0.5 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.5 \\ 2 & 1.75 \\ 1.5 & 2.25 \end{pmatrix},$$

and the mixing proportions

$$\boldsymbol{\pi} = (0.3, 0.3, 0.4), \quad \boldsymbol{\rho}^\mu = (0.3, 0.5, 0.2), \quad \boldsymbol{\rho}^\Sigma = (0.4, 0.6).$$

Table 3 shows the average error of the estimates over the 50 datasets, and the average ARI values over the 50 datasets for each partition are shown in Table 4. Again, we obtain very good classification performance for all three partitions. The progression of the parameter estimates is shown in Figure 3. Similar to Simulation 1, a burn-in period of 20 iterations is

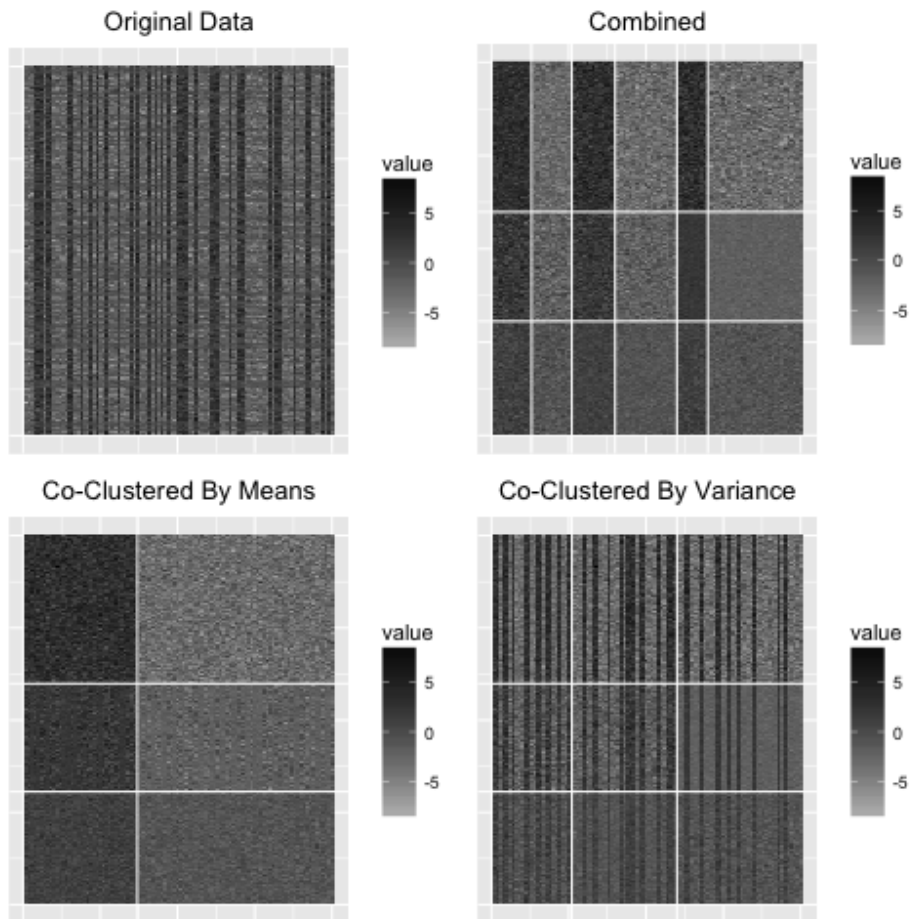


Figure 2: Estimated co-clustering solution for one of the fifty datasets from Simulation 1.

still sufficient to obtain a stable chain. Finally, Figure 4 displays the co-clustering solutions for one of the 50 datasets. Unlike in the first simulation, there is very little spatial separation between blocks.

Table 3: Average error (and standard deviation) of the estimates over the 50 datasets for Simulation 2.

$\overline{\Delta\mu}$	$\overline{\Delta\Sigma}$	$\overline{\Delta\pi}$	$\overline{\Delta\rho^\mu}$	$\overline{\Delta\rho^\Sigma}$
0.15 (0.50)	0.085 (0.046)	1.29e-15 (3.91e-16)	0.015 (0.088)	0.0079 (0.0054)

4.2 Simulation 3

In this simulation, the performance of the ICL–BIC selection criterion is considered. Again, 50 datasets are simulated with $n = 2000$, $p = 500$, $G = L^\mu = L^\Sigma = 3$,

Table 4: Average ARI (and standard deviation) for the row ($\overline{\text{ARI}}_r$), column by means ($\overline{\text{ARI}}_{c\mu}$), and column by variances ($\overline{\text{ARI}}_{c\Sigma}$) partitions over the 50 datasets for Simulation 2.

$\overline{\text{ARI}}_r$	$\overline{\text{ARI}}_{c\mu}$	$\overline{\text{ARI}}_{c\Sigma}$
1.00 (0.00)	0.98 (0.080)	0.96 (0.018)

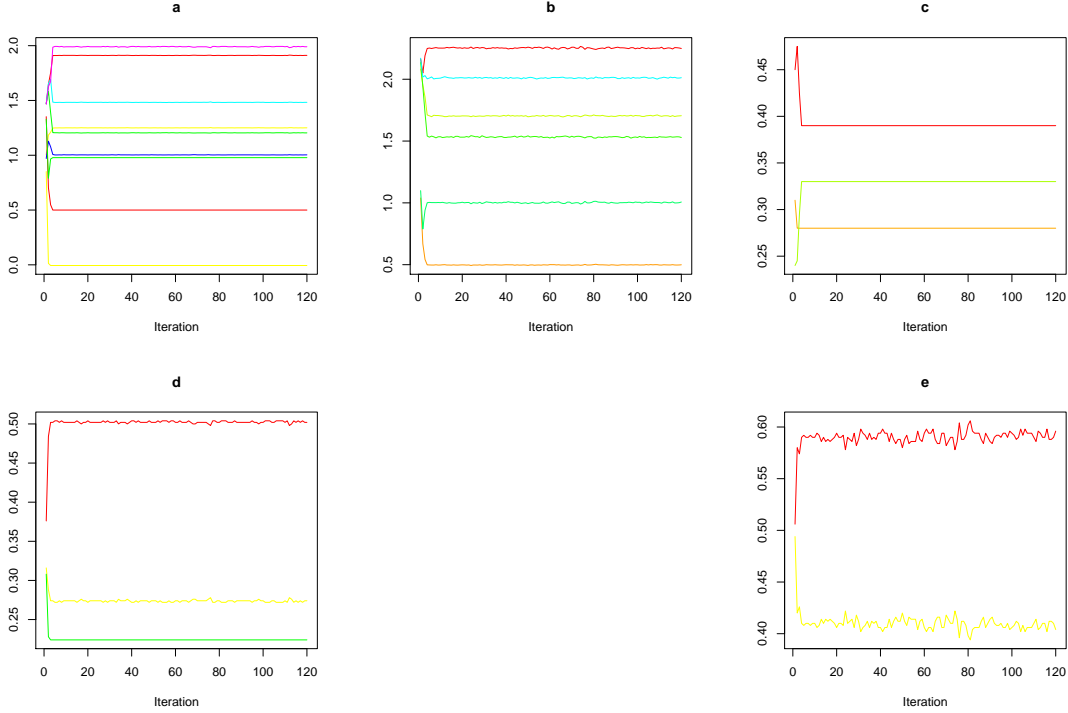


Figure 3: Simulation 2 SEM algorithm parameter estimation progression for one dataset for (a) the mean parameters $\mu_{gl\mu}$, (b) the variance parameters $\sigma_{gl\Sigma}^2$, (c) the row mixing proportions π_g , (d) the column by means mixing proportions $\rho_{l\mu}^\mu$, and (e) the column by variances mixing proportions $\rho_{l\Sigma}^\Sigma$.

$$\boldsymbol{\mu} = \begin{pmatrix} 1 & 1.25 & 0 \\ 2 & 1.2 & 1 \\ 1.5 & 1.9 & 0.5 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.5 & 0.25 \\ 2 & 1.75 & 0.5 \\ 1.5 & 2.25 & 1 \end{pmatrix},$$

and mixing proportions

$$\boldsymbol{\pi} = (0.3, 0.3, 0.4), \quad \boldsymbol{\rho}^\mu = (0.3, 0.4, 0.3), \quad \boldsymbol{\rho}^\Sigma = (0.4, 0.3, 0.3).$$

An exhaustive search is performed considering each of combination of $G, L^\mu, L^\Sigma \in \{2, 3, 4\}$. In Table 5, the number of times each value of G, L^μ and L^Σ is chosen by the ICL-BIC is

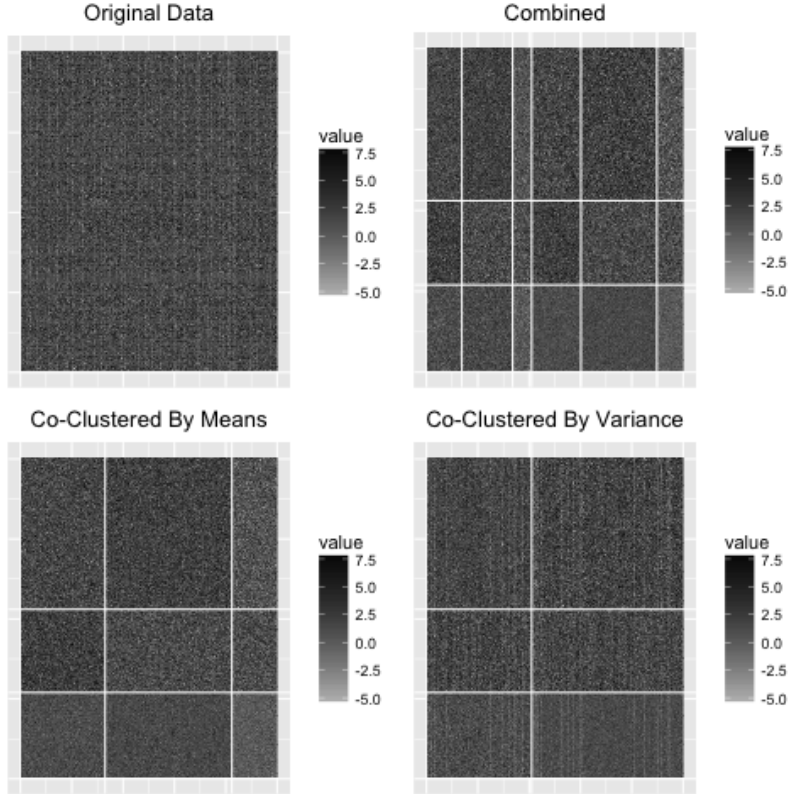


Figure 4: Estimated co-clustering solution for one of the fifty datasets from Simulation 2.

displayed. For the vast majority of the datasets, the correct model is chosen by the ICL–BIC.

Table 5: Frequency of the number of row-clusters, column-clusters by means, and column-clusters by variances chosen by the ICL–BIC over the 50 simulated datasets when using the exhaustive search in Simulation 3.

	2	3	4
G	0	49	1
L^μ	0	48	2
L^Σ	0	48	2

4.3 Simulation 4

In the last simulation, the performance of the non-exhaustive search algorithm described in Section 3.3 is addressed. In all, 25 datasets are simulated according to the parameters

$n = 100, p = 200, G = L^\Sigma = 3, L^\mu = 4,$

$$\boldsymbol{\mu} = \begin{pmatrix} 1 & -0.25 & 0.3 & -1 \\ 1.25 & 0 & 0.1 & -0.3 \\ 0.5 & -1 & 0 & 0.1 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.5 & 0.25 \\ 2 & 1.75 & 0.5 \\ 1.5 & 2.25 & 1 \end{pmatrix},$$

and

$$\boldsymbol{\pi} = (0.3, 0.3, 0.4), \quad \boldsymbol{\rho}^\mu = (0.2, 0.3, 0.25, 0.25), \quad \boldsymbol{\rho}^\Sigma = (0.5, 0.25, 0.25).$$

The initial values are taken to be $(G_1, L_1^\mu, L_1^\Sigma) = (1, 1, 1)$ and the maximum values for all three are set to five. In Table 6, the number of times each value of G , L^μ and L^Σ is chosen by the ICL–BIC is shown. Notice that the procedure performs quite well for choosing the correct model.

Table 6: Frequency of the number of row-clusters, column-clusters by means, and column-clusters by variances chosen by the ICL–BIC over the 25 simulated datasets when using the non-exhaustive search method for Simulation 4.

	2	3	4
G	0	24	1
L^μ	0	0	25
L^Σ	1	24	0

5 Real Data Analyses

5.1 Comparing Parameter-Wise and Traditional Co-Clustering Under Similar Conditions

A subset of the Jester dataset used by Goldberg et al. (2001) is used to compare parameter-wise co-clustering and traditional co-clustering. The data consist of 100 jokes rated on a “continuous” scale from -10 to 10 . A total of 7200 users rated all 100 jokes, and a random sample of 2000 of these users is considered herein.

The non-exhaustive search algorithm is performed for traditional co-clustering with the number of row-clusters ranging from one to 25 and the number of column-clusters ranging from one to seven. This results in choosing seven row-clusters and three column-clusters and the resultant ICL–BIC is -569487.0 . With these values for G and L , the total number of free parameters is 50. In the next section, the non-exhaustive search algorithm is used for the proposed parameter-wise method; however, it is interesting to consider the performance of

the parameter-wise method under similar conditions to the results obtained with traditional co-clustering. Specifically, the parameter-wise method is performed on this dataset with $G = 7, L^\mu = L^\Sigma = 3$. Under this model, the ICL-BIC is -569010.4 , and the total number of free parameters is 52. Note that the ICL-BIC values for both traditional and parameter-wise co-clustering are quite similar, with a slightly higher value obtained when using parameter-wise co-clustering. In Figure 5, the original data (left panel) and the traditional co-clustering solution (right panel), are shown, and the co-clustering solutions for parameter-wise co-clustering are displayed (Figure 6) in the same format as the simulations. Notice that a total of seven combined column-clusters are obtained when using parameter-wise co-clustering.

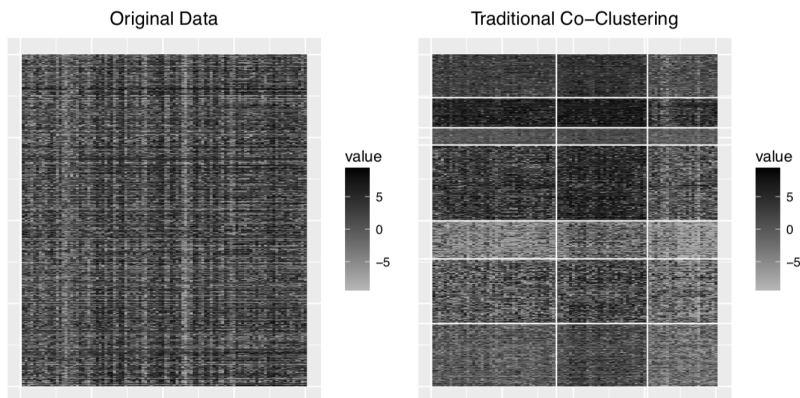


Figure 5: Traditional co-clustering results for the Jester data.

In Table 7, we show a classification table comparing the column-clusters by means and column-clusters by variances found using parameter-wise co-clustering and the column-clusters found using traditional co-clustering. There is almost perfect agreement between the column-clusters from traditional co-clustering and the column-clusters by means from parameter-wise co-clustering. This, however, is not true for the column-clusters by variances. This result is somewhat perceptible in the images of the co-clustering solutions. In Table 8, the classification table comparing row-clusters from traditional and parameter-wise co-clustering is displayed. It is clear that the row-clusters found by both of these methods are quite comparable — the ARI when comparing these two partitions is 0.86.

5.2 Further Analysis with Parameter-Wise Co-Clustering

The non-exhaustive search algorithm is now performed for parameter-wise co-clustering. The range of values was one to 25 row-clusters, and one to seven column-clusters by means and column-clusters by variances resulting in the ICL-BIC choosing a model with 17 row-clusters, six column-clusters by means, and four column-clusters by variances. The resulting

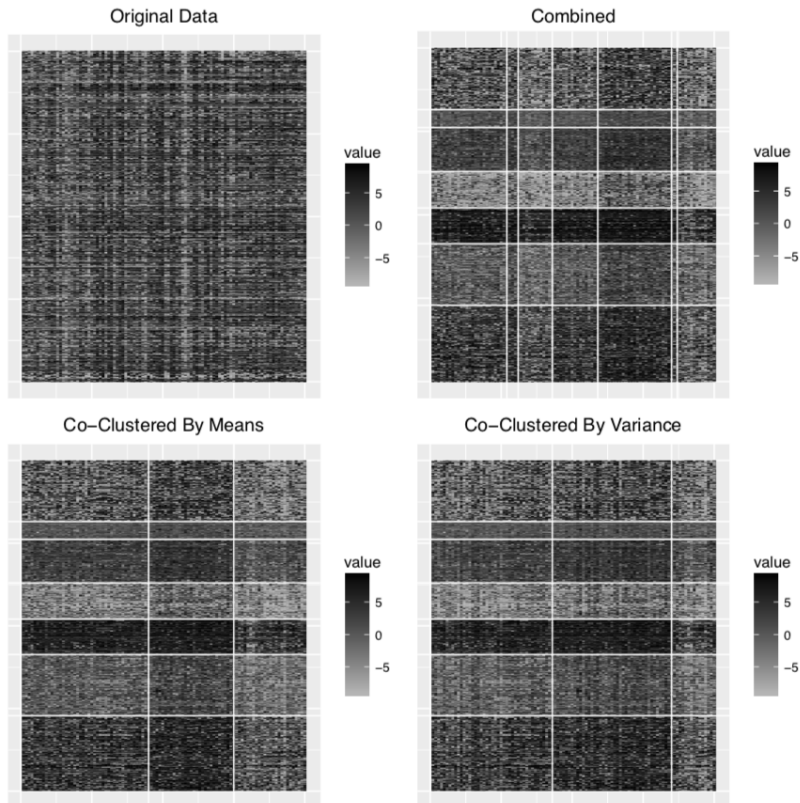


Figure 6: Parameter-wise co-clustering results for the Jester dataset under similar conditions to the traditional co-clustering solution.

ICL-BIC is -561099.0 and a total of 15 combined column-clusters are obtained. Notice that there is significant improvement in the ICL-BIC in this case. In Figure 7, we show the parameter-wise co-clustering solution. Because more row-clusters are obtained, it is far more difficult to visualize the row-clusters. Moreover, the combined co-clustering solution is very difficult to interpret in this scenario, which displays the benefit of visualizing the column-clusters by means and column-clusters by variances separately.

Finally, the exhaustive search algorithm is performed for both traditional and parameter-wise co-clustering. For each value of $G \in \{1, 2, \dots, 25\}$, the maximum ICL-BIC over all values of L for traditional co-clustering, and L^μ and L^Σ for parameter-wise co-clustering is considered. In Figure 8, we display a plot of this maximum ICL-BIC against G . For both traditional and parameter-wise co-clustering, the ICL-BIC begins to plateau around $G = 10$. Moreover, the ICL-BIC for parameter-wise co-clustering is oftentimes, if only very slightly, higher than traditional co-clustering. Finally, we note that it is very computationally expensive to run the exhaustive search with parameter-wise co-clustering taking around 24 hours using 25 1200MHz cores running continuously.

Table 7: Classification table comparing the column-clusters by means and column-clusters by variances for parameter-wise co-clustering and column-clusters from traditional co-clustering for the Jester dataset.

Traditional	Means			Variances		
	1	2	3	1	2	3
1	43	0	1	28	14	2
2	2	30	0	4	28	0
3	0	0	24	11	0	13

Table 8: Classification table comparing row-clusters for parameter-wise and traditional co-clustering.

Parameter-Wise	Traditional						
	1	2	3	4	5	6	7
1	427	10	1	0	3	0	16
2	0	350	0	9	0	0	11
3	18	0	180	0	16	0	0
4	0	0	0	216	0	0	3
5	10	11	0	0	241	1	0
6	0	5	0	0	0	103	0
7	2	3	0	4	0	0	360

6 Discussion

A parameter-wise co-clustering algorithm was developed for high-dimensional data. This parameter-wise method allowed for two partitions of the columns based on both means and variances, as well as a combined co-clustering solution. This, in essence, provides more flexibility than traditional co-clustering, while maintaining the high degree of parsimony inherent to traditional co-clustering. An SEM Gibbs algorithm was used for parameter estimation, and evaluated by two simulations. An ICL-BIC criterion, as well as a non-exhaustive search algorithm, were developed for model selection.

A subset of the Jester dataset was considered for comparison purposes between traditional and parameter-wise co-clustering. After applying traditional co-clustering to the data, parameter-wise co-clustering was performed using similar parameters, i.e., same G and $L^\mu = L^\Sigma = L$. This resulted in similar row-clusters between the two methods. Furthermore, the column-clusters by means using parameter-wise co-clustering were almost identical to

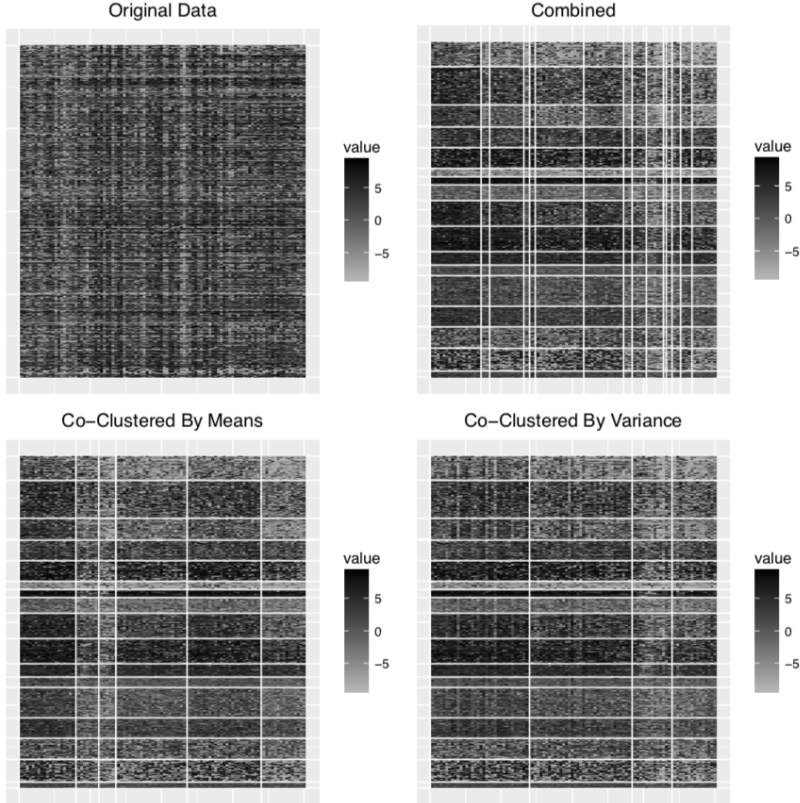


Figure 7: Parameter-wise co-clustering results for the Jester data after performing the non-exhaustive search algorithm.

the column-clusters from traditional co-clustering. This was not true, however, when comparing the column-clusters by variances and the column-clusters obtained from traditional co-clustering. Parameter-wise co-clustering also had a marginally higher ICL-BIC in this case. Using the non-exhaustive search algorithm for parameter-wise co-clustering resulted in far more row-clusters, and many more combined column-clusters, which displayed the utility of considering the co-clustering by means, and co-clustering by variances separately from the combined co-clustering solution.

Although this method only considered the use of the Gaussian distribution, it can be extended in various ways. One example would be to use other continuous distributions with more than one parameter. For example, one could consider the skew- t distribution and cluster columns based on location, scale, concentration and skewness. This could also be extended to data that cannot be considered a realization of a continuous random variable such as ordinal data where the columns could be partitioned according to mode and precision. The number of free parameters in each of these cases will not depend on the dimensionality of the data thus preserving the parsimony inherent to co-clustering.

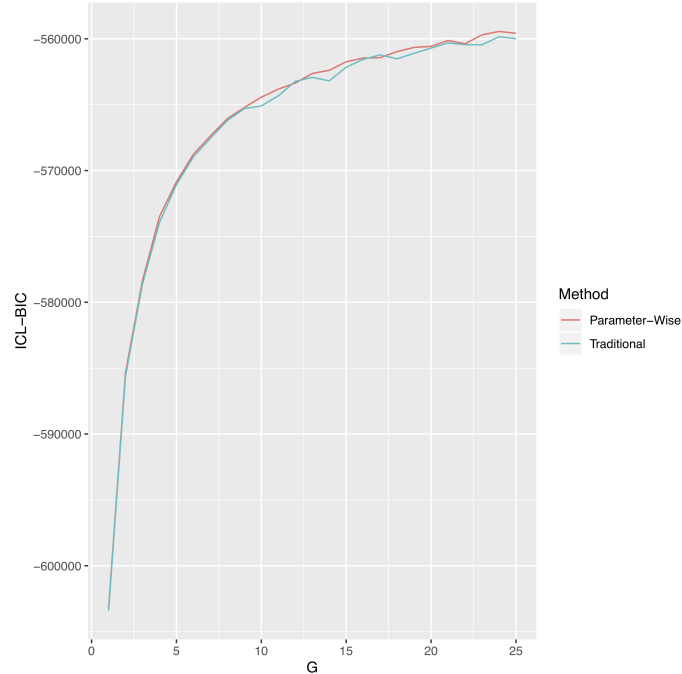


Figure 8: Maximum ICL-BIC over L for traditional co-clustering (turquoise), and L^μ and L^Σ for parameter-wise co-clustering (red) for each value of G , against G .

References

- Baum, L. E., Petrie, T., Soules, G. and Weiss, N. (1970), ‘A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains’, *Annals of Mathematical Statistics* **41**, 164–171.
- Biernacki, C., Celeux, G. and Govaert, G. (2000), ‘Assessing a mixture model for clustering with the integrated completed likelihood’, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(7), 719–725.
- Biernacki, C. and Maugis, C. (2017), High-dimensional clustering, *in* ‘Choix de modèles et agrégation, Sous la direction de J-J. Dreesbeke, G. Saporta, C. Thoma-Agnan Edition: Technip.’.
- Bouveyron, C. and Brunet-Saumard, C. (2014), ‘Model-based clustering of high-dimensional data: A review’, *Computational Statistics and Data Analysis* **71**, 52–78.
- Bouveyron, C., Girard, S. and Schmid, C. (2007), ‘High-dimensional data clustering’, *Computational Statistics and Data Analysis* **52**(1), 502–519.

- Brault, V., Keribin, C. and Mariadassou, M. (2017), ‘Consistency and asymptotic normality of latent blocks model estimators’. arXiv preprint arXiv:1704.06629.
- Ghahramani, Z. and Hinton, G. E. (1997), The EM algorithm for factor analyzers, Technical Report CRG-TR-96-1, University of Toronto, Toronto, Canada.
- Goldberg, K., Roeder, T., Gupta, D. and Perkins, C. (2001), ‘Eigentaste: A constant time collaborative filtering algorithm’, *Information Retrieval* **4**(2), 133–151.
- Hartigan, J. A. (1972), ‘Direct clustering of a data matrix’, *Journal of the American statistical association* **67**(337), 123–129.
- Hubert, L. and Arabie, P. (1985), ‘Comparing partitions’, *Journal of Classification* **2**(1), 193–218.
- Jacques, J. and Biernacki, C. (2018), ‘Model-based co-clustering for ordinal data’, *Computational Statistics & Data Analysis* **123**, 101–115.
- McLachlan, G. and Peel, D. (2000), Mixtures of factor analyzers, in ‘In Proceedings of the Seventeenth International Conference on Machine Learning’, Morgan Kaufmann, San Francisco, pp. 599–606.
- McNicholas, P. D. (2016), ‘Model-based clustering’, *Journal of Classification* **33**(3), 331–373.
- McNicholas, P. D. and Murphy, T. B. (2008), ‘Parsimonious Gaussian mixture models’, *Statistics and Computing* **18**(3), 285–296.
- Meynet, C. and Maugis-Rabusseau, C. (2012), A sparse variable selection procedure in model-based clustering, Research report.
- Nadif, M. and Govaert, G. (2010), Model-based co-clustering for continuous data, in ‘Machine Learning and Applications (ICMLA), 2010 Ninth International Conference on’, IEEE, pp. 175–180.
- Pan, W. and Shen, X. (2007), ‘Penalized model-based clustering with application to variable selection’, *Journal of Machine Learning Research* **8**(May), 1145–1164.
- Pledger, S. and Arnold, R. (2014), ‘Multivariate methods using mixtures: Correspondence analysis, scaling and pattern-detection’, *Computational Statistics & Data Analysis* **71**, 241–261.

- Robert, V. (2017), Coclustering for the analysis of pharmacovigilance massive datasets, PhD thesis, Université Paris-Saclay. Hal preprint: tel-01806330.
- Schwarz, G. (1978), ‘Estimating the dimension of a model’, *The Annals of Statistics* **6**(2), 461–464.
- Scott, A. J. and Symons, M. J. (1971), ‘Clustering methods based on likelihood ratio criteria’, *Biometrics* **27**, 387–397.
- Tipping, M. E. and Bishop, C. M. (1999), ‘Mixtures of probabilistic principal component analysers’, *Neural Computation* **11**(2), 443–482.
- Wolfe, J. H. (1965), A computer program for the maximum likelihood analysis of types, Technical Bulletin 65-15, U.S. Naval Personnel Research Activity.
- Zhou, H., Pan, W. and Shen, X. (2009), ‘Penalized model-based clustering with unconstrained covariance matrices’, *Electronic Journal of Statistics* **3**, 1473.