Chapter 5

Co-Clustering of Continuous Data

In this chapter, we focus on the co-clustering of an $n \times d$ data matrix $\mathbf{x} = (x_{ij})$ corresponding to the observations of a set J of d continuous variables on a set I of n individuals. As for the principal component analysis (PCA), we associate with the set of individuals weights $\mathbf{p} = (p_1, \ldots, p_n)$ and with the set of variables weights $\mathbf{q} = (q_1, \ldots, q_d)$. Moreover, we consider the data matrix $\mathbf{x}$ to be "column-centered", i.e. the means of the columns of $\mathbf{x}$ are equal to 0. Note that, if it is not true, it is easy to modify the initial table to obtain this property. Finally, the initial data that are defined by a "column-centered" matrix $\mathbf{x}$ and two vectors $\mathbf{p}$ and $\mathbf{q}$ will be represented by the triple $(\mathbf{x}, \mathbf{p}, \mathbf{q})$.

To analyze this type of data, we can use PCA that summarizes data by means of new axes. In this chapter, the aim is to summarize the data by means of clusters of the individuals and the variables. There are different ways to attempt this block clustering objective. The simple one consists of clustering the set of variables and, using the data obtained by replacing each variable with the mean of its cluster, applying a clustering algorithm to the set of

Co-Clustering: Models, Algorithms and Applications Gérard Govaert and Mohamed Nadif

individuals. For cluster variables, we can use partitioning methods such as k-means, or hierarchical methods such as the well-known SAS Varclus procedure, which is a type of oblique component analysis related to multiple group factor analysis [HAR 76]. The k-means applied on centered variables and Varclus are used as variable-reduction methods where the variables of the same cluster are correlated as much as possible, and two variables belonging to different clusters are as uncorrelated as possible.

In order to obtain homogeneous blocks, there are more adapted methods that consist of clustering both sets *simultaneously*. Most methods used in this context can be grouped into two families: metric methods and model-based methods. In the first family, which aims to optimize an objective function, we can cite, for instance, the works of Hartigan [HAR 72], Bock [BOC 79] and Govaert [GOV 83, GOV 95]. In the second family, two approaches can be considered: the latent block model with Gaussian distributions and the mixture model with conditional independent Gaussian distributions more adapted to continuous data.

This chapter is organized as follows. Section 5.1 describes, the metric methods, which can be viewed as a minimization of a loss information function. We describe the CROEUC algorithm used for this purpose and, before concluding, we present an example. In section 5.2, we study the latent block model with Gaussian distributions and associated algorithms Gaussian LBCEM and LBVEM that we illustrate by an example in section 5.3. In section 5.4, we present a Gaussian block mixture model and derive two algorithms that we evaluate in section 5.5.

5.1. Metric approach

Two geometrical representations can be associated with continuous data:

– A geometrical representation of the individuals by a set of n points of $\mathbb{R}^d$ where the coordinates of the n points are the rows of the data matrix $\mathbf{x}$, the p_i are the weights of the point and the q_j can be used to define the Euclidean metric $\mathbf{d}^2(i,i') = \sum_j q_j(x_{ij} - x_{i'j})^2$.

– A geometrical representation of the variables by a set of d points of $\mathbb{R}^n$ where the coordinates of the n points are the columns of the data matrix $\mathbf{x}$, the q_j are the weights of the point and the p_i can be used to define the Euclidean metric $\mathbf{d}^2(j,j') = \sum_i p_i(x_{ij} - x_{ij'})^2$.

In the following, and only to simplify the notation, we assume that $p_i = \frac{1}{n}$ for all i and $q_j = 1$ for all j.

5.1.1. *Measure of information*

The information measure we would like to preserve is the following:

$$\mathcal{I}(\mathbf{x}, \mathbf{p}, \mathbf{q}) = \sum_{i,j} p_i q_j x_{ij}^2 = \frac{1}{n} \sum_{i,j} x_{ij}^2.$$

With the geometrical representations, this information represents in $\mathbb{R}^d$ the inertia of the set I relative to the center of gravity and in $\mathbb{R}^n$ the inertia of the set J relative to the origin. Let us note that this information measure is the measure used by PCA.

5.1.2. *Summarized data associated with partitions*

As was seen in section 2.1 of Chapter 2, the continuous data matrix can be summarized by a smaller continuous matrix by associating each block with its mean (see Figure 5.1). More precisely, using a partition $\mathbf{z}$ of I and a partition $\mathbf{w}$ of J, the initial data are summarized by two sets

of weights $\mathbf{p^z} = (p_1^{\mathbf{z}}, \ldots, p_g^{\mathbf{z}})$ and $\mathbf{q^w} = (q_1^{\mathbf{w}}, \ldots, q_m^{\mathbf{w}})$ and a $g \times m$ matrix $\mathbf{x^{zw}} = (x_{k\ell}^{\mathbf{zw}})$ defined by

$$p_k^{\mathbf{z}} = \frac{\sum_i z_{ik}}{n} = \frac{z_{.k}}{n}, \qquad q_\ell^{\mathbf{w}} = \sum_j w_{j\ell} = w_{.\ell}$$

and

$$x_{k\ell}^{\mathbf{zw}} = \frac{\sum_{i,j} z_{ik} w_{j\ell} p_i q_j x_{ij}}{\sum_{i,j} z_{ik} w_{j\ell} p_i q_j} = \frac{\sum_{i,j} z_{ik} w_{j\ell} x_{ij}}{z_{.k} w_{.\ell}}.$$

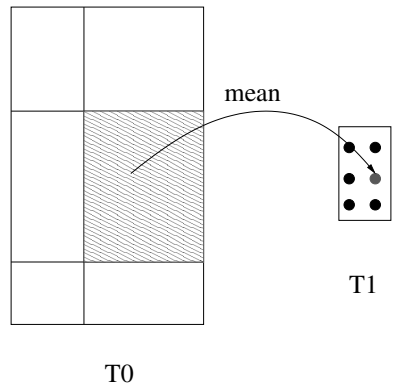

Figure 5.1. *Aggregation of data matrix in summary matrix using co-clustering*

We have, thus, associated the two partitions $\mathbf{z}$ and $\mathbf{w}$ with a new triple $(\mathbf{x^{zw}}, \mathbf{p^z}, \mathbf{q^w})$ which has the same structure as the initial data $(\mathbf{x}, \mathbf{p}, \mathbf{q})$ and therefore we can define the information measure

$$\mathcal{I}(\mathbf{x^{zw}}, \mathbf{p^z}, \mathbf{q^w}) = \sum_{k,\ell} p_k^{\mathbf{z}} q_\ell^{\mathbf{w}} (x_{k\ell}^{\mathbf{zw}})^2 = \frac{1}{n} \sum_{k,\ell} z_{.k} w_{.\ell} (x_{k\ell}^{\mathbf{zw}})^2. \qquad [5.1]$$

We have illustrated this summary matrix in Table 5.1. Starting from the data $\mathbf{x}$ with weights $p_i = 1/4$, $q_j = 1$ and partitions $\mathbf{z} = (1, 1, 2, 2)$ and $\mathbf{w} = (1, 1, 2)$, we obtain the summary $\mathbf{x^{zw}}$ with weights $\mathbf{p^z} = (1/2, 1/2)$ and $\mathbf{q^w} = (2, 1)$.

$$\mathbf{x} = \begin{pmatrix} 1 & 2 & 8 \\ 2 & 1 & 7 \\ 2 & 4 & 7 \\ 4 & 4 & 6 \end{pmatrix} \qquad \mathbf{x}^{\mathbf{zw}} = \begin{pmatrix} 1.5 & 7.5 \\ 3.5 & 6.5 \end{pmatrix}$$

Table 5.1. *Example of summary*

As in Chapters 3 and 4, intermediate matrices $\mathbf{x}^{\mathbf{w}} = (x_{i\ell}^{\mathbf{w}})$ of size $(n \times m)$ and $\mathbf{x}^{\mathbf{z}} = (x_{kj}^{\mathbf{z}})$ of size $(g \times m)$ can be defined but this time we have

$$x_{i\ell}^{\mathbf{w}} = \frac{\sum_{j,\ell} w_{j\ell} q_j x_{ij}}{\sum_{j,\ell} w_{j\ell} q_j} = \frac{\sum_{j,\ell} w_{j\ell} x_{ij}}{w_{\cdot\ell}} \quad \text{and}$$

$$x_{kj}^{\mathbf{z}} = \frac{\sum_{i,k} z_{ik} p_i x_{ij}}{\sum_{i,k} p_i z_{ik}} = \frac{\sum_{i,k} z_{ik} x_{ij}}{z_{\cdot k}}$$

and with the data defined in Table 5.1, these intermediate matrices become

$$\mathbf{x}^{\mathbf{z}} = \begin{pmatrix} 1.5 & 1.5 & 7.5 \\ 3 & 4 & 6.5 \end{pmatrix} \quad \text{and} \quad \mathbf{x}^{\mathbf{w}} = \begin{pmatrix} 1.5 & 8 \\ 1.5 & 7 \\ 3 & 7 \\ 4 & 6 \end{pmatrix}.$$

In the following, we will denote $(\mathbf{x}^{\mathbf{w}}, \mathbf{p}, \mathbf{q}^{\mathbf{w}})$ as the triple obtained when $\mathbf{z}$ is the singleton partition and $(\mathbf{x}^{\mathbf{z}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q})$ as the triple obtained when $\mathbf{w}$ is the singleton partition. Hence, we obtain the associated measures of association as follows

$$\mathcal{I}(\mathbf{x}^{\mathbf{z}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q}) = \frac{1}{n} \sum_{k,j} z_{\cdot k} (x_{kj}^{\mathbf{z}})^2 \text{ and } \mathcal{I}(\mathbf{x}^{\mathbf{w}}, \mathbf{p}, \mathbf{q}^{\mathbf{w}}) = \frac{1}{n} \sum_{i,\ell} w_{\cdot\ell} (x_{i\ell}^{\mathbf{w}})^2.$$

REMARK 5.1.– When $\mathbf{w}$ is the partition of singletons, this criterion can be expressed as the loss of information due to the partition $\mathbf{z}$ and, by using the Huygens theorem, it can be shown that

$$\mathcal{I}(\mathbf{x}, \mathbf{p}, \mathbf{q}) - \mathcal{I}(\mathbf{x}^{\mathbf{z}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q}) = \frac{1}{n} \mathrm{W}(\mathbf{z}|J)$$

where $\mathrm{W}(\mathbf{z}|J) = \sum_{i,k} z_{ik} \sum_j (x_{ij} - x_{kj}^{\mathbf{z}})^2$ is the intra class inertia, or within-group sum of squares, minimized by the classical k-means algorithm. Similarly, when $\mathbf{z}$ is the partition of singletons, we have

$$\mathcal{I}(\mathbf{x}, \mathbf{p}, \mathbf{q}) - \mathcal{I}(\mathbf{x}^{\mathbf{w}}, \mathbf{p}, \mathbf{q}^{\mathbf{w}}) = \frac{1}{n} \mathrm{W}(\mathbf{w}|I)$$

where $\mathrm{W}(\mathbf{w}|I) = \sum_{j,\ell} w_{j\ell} \sum_i (x_{ij} - x_{i\ell}^{\mathbf{w}})^2$.

5.1.3. *Objective function*

It can be easily proved that the measure of information associated with the summarized data $(\mathbf{x}^{\mathbf{zw}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q}^{\mathbf{w}})$ is smaller than the information associated with the initial data, i.e. grouping rows and columns leads to a loss of information. Therefore, the objective will be to minimize the objective function

$$\begin{aligned} \mathrm{W}(\mathbf{z}, \mathbf{w}) &= \mathcal{I}(\mathbf{x}, \mathbf{p}, \mathbf{q}) - \mathcal{I}(\mathbf{x}^{\mathbf{zw}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q}^{\mathbf{w}}) \\ &= \frac{1}{n} \sum_{i,j,k,\ell} z_{ik} w_{j\ell} (x_{ij} - x_{k\ell}^{\mathbf{zw}})^2 \end{aligned} \qquad [5.2]$$

or, equivalently, to maximize the measure of information $\mathcal{I}(\mathbf{x}^{\mathbf{zw}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q}^{\mathbf{w}})$.

Note that the objective function defined in equation [5.2] can also be written as

$$||\mathbf{x} - \mathbf{y}||^2 = \mathrm{trace}\left((\mathbf{x} - \mathbf{y})(\mathbf{x} - \mathbf{y})^t\right), \qquad [5.3]$$

where $\mathbf{y} = \mathbf{z}(\mathbf{x}^{\mathbf{zw}})\mathbf{w}^t$ can be viewed as a reconstructed matrix based on the cluster structures. For instance, in the previous example, the matrix $\mathbf{y}$ is

$$\begin{pmatrix} 1.5 & 1.5 & 7.5 \\ 1.5 & 1.5 & 7.5 \\ 3.5 & 3.5 & 6.5 \\ 3.5 & 3.5 & 6.5 \end{pmatrix}.$$

The clustering problem can therefore be formulated as a matrix approximation problem where the clustering aim is to minimize the approximation error between the original data x and the reconstructed matrix y based on the cluster structures. This approximation can be solved by an iterative alternating least-squares optimization procedure (see, for instance, [BAI 97, VIC 01, CHO 04, LI 05, ROS 09, LAB 11a]). These algorithms are equivalent and consist of using the principle of a double k-means. A version based on update rules is presented in algorithm 5.1. Furthermore, we recommend another version called CROEUC [GOV 83, GOV 95], which is based on the use of reduced intermediate matrices $\mathbf{x}^{\mathbf{w}}$ and $\mathbf{x}^{\mathbf{z}}$ and therefore requires less computation.

Algorithm 5.1 Double k-means

Input: x, g, m
Initialization: z, w, $x_{k\ell}^{\mathbf{zw}} = \sum_{i,j} \frac{z_{ik} w_{j\ell} x_{ij}}{z_{.k} w_{.\ell}}$
repeat
  step 1. $z_i = \arg\min_k \sum_{j,\ell} w_{j\ell}(x_{ij} - x_{k\ell}^{\mathbf{zw}})^2$
  step 2. $w_j = \arg\min_\ell \sum_{i,k} z_{ik}(x_{ij} - x_{k\ell}^{\mathbf{zw}})^2$
  step 3. $x_{k\ell}^{\mathbf{zw}} = \sum_{i,j} \frac{z_{ik} w_{j\ell} x_{ij}}{z_{.k} w_{.\ell}}$
until convergence
return z, w

5.1.4. CROEUC *algorithm*

Using the strategy described in section 2.4.1 of Chapter 2, the CROEUC algorithm consists of computing, for a fixed partition of J, the best partition of I, and for a fixed partition J, the best partition of I. For this, it can be noted that following the Huygens theorem, we have

$$\mathcal{I}(\mathbf{x}^{\mathbf{w}}, \mathbf{p}, \mathbf{q}^{\mathbf{w}}) - \mathcal{I}(\mathbf{x}^{\mathbf{zw}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q}^{\mathbf{w}}) = \mathrm{W}(\mathbf{z}|\mathbf{w}),$$

where

$$\mathrm{W}(\mathbf{z}|\mathbf{w}) = \sum_{i,k} \sum_{\ell} w_{.\ell}(x_{i\ell}^{\mathbf{w}} - x_{k\ell}^{\mathbf{zw}})^2$$

is the intra class inertia criterion for the partition $\mathbf{z}$ when the data are the triple $(\mathbf{x}^{\mathbf{w}}, \mathbf{p}, \mathbf{q}^{\mathbf{w}})$ and

$$\mathcal{I}(\mathbf{x}^{\mathbf{z}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q}) - \mathcal{I}(\mathbf{x}^{\mathbf{zw}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q}^{\mathbf{w}}) = \mathrm{W}(\mathbf{w}|\mathbf{z}),$$

where

$$\mathrm{W}(\mathbf{w}|\mathbf{z}) = \sum_{j,\ell} w_{j\ell} \sum_{k} z_{.k} (x^{\mathbf{z}}_{kj} - x^{\mathbf{zw}}_{k\ell})^2$$

is the intraclass inertia criterion for the partition $\mathbf{w}$ when the data are the triple $(\mathbf{x}^{\mathbf{z}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q})$. Therefore, the minimization of the objective function [5.2] can be performed by alternating the k-means algorithm on the triple $(\mathbf{x}^{\mathbf{w}}, \mathbf{p}, \mathbf{q}^{\mathbf{w}})$ and on the triple $(\mathbf{x}^{\mathbf{z}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q})$ and we obtain an algorithm optimizing $\mathcal{I}(\mathbf{x}^{\mathbf{zw}}, \mathbf{p}^{\mathbf{z}}, \mathbf{q}^{\mathbf{w}})$. The principal steps are described in algorithm 5.2.

Algorithm 5.2 CROEUC

input: $\mathbf{x}$, g, m
initialization: $\mathbf{z}$, $\mathbf{w}$
repeat
  $x^{\mathbf{w}}_{i\ell} = \frac{1}{w_{.\ell}} \sum_j w_{j\ell} x_{ij}$, $x^{\mathbf{zw}}_{k\ell} = \frac{1}{z_{.k}} \sum_i z_{ik} x^{\mathbf{w}}_{i\ell}$
  repeat
    step 1. $z_i = \arg\min_k \sum_\ell w_{.\ell} (x^{\mathbf{w}}_{i\ell} - x^{\mathbf{zw}}_{k\ell})^2$
    step 2. $x^{\mathbf{zw}}_{k\ell} = \frac{\sum_i z_{ik} x^{\mathbf{w}}_{i\ell}}{z_{.k}}$
  until convergence
  $x^{\mathbf{z}}_{kj} = \frac{1}{z_{.k}} \sum_i z_{ik} x_{ij}$, $x^{\mathbf{zw}}_{k\ell} = \frac{1}{w_{.\ell}} \sum_j z_{j\ell} x^{\mathbf{z}}_{kj}$
  repeat
    step 3. $w_j = \arg\min_\ell \sum_k z_{.k} (x^{\mathbf{z}}_{kj} - x^{\mathbf{zw}}_{k\ell})^2$
    step 4. $x^{\mathbf{zw}}_{k\ell} = \frac{\sum_j w_{j\ell} x^{\mathbf{z}}_{kj}}{w_{.\ell}}$
  until convergence
until convergence
return $\mathbf{z}$, $\mathbf{w}$

REMARK 5.2.– Steps 1 and 2 can be seen as a k-means applied on $\mathbf{x}^{\mathbf{w}}$ using the Euclidean distance weighted by $(w_{.1}, \ldots, w_{.m})$

and the mean values of each block. Similarly, steps 3 and 4 can be seen as k-means applied on $\mathbf{x}^{\mathbf{z}}$ using the Euclidean distance weighted by $(z_{.1}, \ldots, z_{.g})$.

As we have seen in the previous chapters, co-clustering can be embedded in the probabilistic approach. In this chapter, two probabilistic models are studied and we begin by considering the latent block model.

5.2. Gaussian latent block model

5.2.1. *The model*

Using the latent block model described in section 2.3 of Chapter 2 in the continuous situation, and assuming that for each block $k\ell$ the values x_{ij} are distributed according to a Gaussian distribution

$$\mathcal{N}(\mu_{k\ell}, \sigma^2_{k\ell}) \qquad \text{with} \quad \mu_{k\ell} \in \mathbb{R} \quad \text{and} \quad \sigma^2_{k\ell} \in \mathbb{R}^+,$$

we obtain the Gaussian latent block model with the following probability density function (pdf) $f(\mathbf{x}; \boldsymbol{\theta})$ taking the form

$$\sum_{(\mathbf{z},\mathbf{w}) \in \mathcal{Z} \times \mathcal{W}} \prod_{i,k} \pi_k^{z_{ik}} \prod_{j,\ell} \rho_\ell^{w_{j\ell}} \prod_{i,j,k,\ell} \times \left(\frac{1}{\sqrt{2\pi\sigma^2_{k\ell}}} \exp - \left\{ \frac{1}{2\sigma^2_{k\ell}} (x_{ij} - \mu_{k\ell})^2 \right\} \right)^{z_{ik} w_{j\ell}} \qquad [5.4]$$

parameterized by $\boldsymbol{\theta} = (\boldsymbol{\pi}, \boldsymbol{\rho}, \boldsymbol{\alpha})$ where $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_g)$, $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_m)$ and $\boldsymbol{\alpha} = (\mu_{11}, \ldots, \mu_{gm}, \sigma^2_{11}, \ldots, \sigma^2_{gm})$. With this model, the complete-data log-likelihood is, up to the constant $-\frac{nd}{2} \log 2\pi$, given by

$$\mathrm{L_C}(\mathbf{z}, \mathbf{w}, \boldsymbol{\theta}) = \sum_{k,\ell} z_{ik} \log \pi_k + \sum_{j,\ell} w_{j\ell} \log \rho_\ell - \frac{1}{2} \sum_{k,\ell} \left(z_{.k} w_{.\ell} \log \sigma^2_{k\ell} + \frac{1}{\sigma^2_{k\ell}} \sum_{i,j} z_{ik} w_{j\ell} (x_{ij} - \mu_{k\ell})^2 \right)$$

and therefore the following fuzzy clustering criterion can be deduced

$$\mathrm{F_C}(\widetilde{\mathbf{z}}, \widetilde{\mathbf{w}}; \boldsymbol{\theta}) = \mathrm{L_C}(\widetilde{\mathbf{z}}, \widetilde{\mathbf{w}}, \boldsymbol{\theta}) + \mathrm{H}(\widetilde{\mathbf{z}}) + \mathrm{H}(\widetilde{\mathbf{w}}). \quad [5.5]$$

5.2.2. *Gaussian* LBVEM *and* LBCEM *algorithms*

Taking into account the definition of the pdf $f(x_{ij}; \mu_{k\ell}, \alpha_{k\ell})$, the variational approximation of the expectation–maximization (EM) algorithm described in section 2.4.1 of Chapter 2 can be completed by the computation of the block parameters $\mu_{k\ell}$ and $\sigma^2_{k\ell}$: for all k, ℓ, the parameters $\mu_{k\ell}$ and $\sigma^2_{k\ell}$ are obtained by the maximization of

$$-\frac{1}{2}\sum_{k,\ell}\left(\widetilde{z}_{.k}\widetilde{\mathrm{w}}_{.\ell}\log\sigma^2_{k\ell} + \frac{1}{\sigma^2_{k\ell}}\sum_{i,j}\widetilde{z}_{ik}\widetilde{\mathrm{w}}_{j\ell}(x_{ij}-\mu_{k\ell})^2\right).$$

We can easily deduce that the parameter $\mu_{k\ell}$, obtained by the minimization of $\sum_{ij}\widetilde{z}_{ik}\widetilde{\mathrm{w}}_{j\ell}(x_{ij}-\mu_{k\ell})^2$, is

$$\mu_{k\ell} = \frac{\sum_{ij}\widetilde{z}_{ik}\widetilde{\mathrm{w}}_{j\ell}x_{ij}}{\widetilde{z}_{.k}\widetilde{\mathrm{w}}_{.\ell}}$$

and that the parameter $\sigma^2_{k\ell}$, obtained by the minimization of

$$\widetilde{z}_{.k}\widetilde{\mathrm{w}}_{.\ell}\log\sigma^2_{k\ell} + \frac{1}{\sigma^2_{k\ell}}\sum_{ij}\widetilde{z}_{ik}\widetilde{\mathrm{w}}_{j\ell}(x_{ij}-\mu_{k\ell})^2,$$

is

$$\sigma^2_{k\ell} = \frac{\sum_{ij}\widetilde{z}_{ik}\widetilde{\mathrm{w}}_{j\ell}(x_{ij}-\mu_{k\ell})^2}{\widetilde{z}_{.k}\widetilde{\mathrm{w}}_{.\ell}} = \frac{\sum_{ij}\widetilde{z}_{ik}\widetilde{\mathrm{w}}_{j\ell}x^2_{ij}}{\widetilde{z}_{.k}\widetilde{\mathrm{w}}_{.\ell}} - \mu^2_{k\ell}.$$

The principal steps of Gaussian LBVEM are described in algorithm 5.3. In a similar way, the maximization of the $\mathrm{L_C}$ criterion can be performed by the Gaussian LBCEM algorithm presented in algorithm 5.4.

Algorithm 5.3 Gaussian LBVEM

input: $\mathbf{x}$, g, m

initialization: $\widetilde{\mathbf{z}}$, $\widetilde{\mathbf{w}}$, $\pi_k = \frac{\widetilde{z}_{.k}}{n}$, $\rho_\ell = \frac{\widetilde{\mathrm{w}}_{.\ell}}{d}$, $\mu_{k\ell} = \frac{x_{k\ell}^{\widetilde{\mathbf{z}}\widetilde{\mathbf{w}}}}{\widetilde{z}_{.k}\widetilde{\mathrm{w}}_{.\ell}}$. $\sigma_{k\ell}^2 = \frac{\sum_{ij}\widetilde{z}_{ik}\widetilde{\mathrm{w}}_{j\ell}x_{ij}^2}{\widetilde{z}_{.k}\widetilde{\mathrm{w}}_{.\ell}} - \mu_{k\ell}^2$

repeat

$x_{i\ell}^{\widetilde{\mathbf{w}}} = \frac{1}{\widetilde{\mathrm{w}}_{.\ell}}\sum_j \widetilde{\mathrm{w}}_{j\ell}x_{ij}$, $u_{i\ell}^{\widetilde{\mathbf{w}}} = \frac{1}{\widetilde{\mathrm{w}}_{.\ell}}\sum_j \widetilde{\mathrm{w}}_{j\ell}x_{ij}^2$

repeat

step 1. $\widetilde{z}_{ik} \propto \pi_k \exp\left(-\frac{1}{2}\sum_\ell \widetilde{\mathrm{w}}_{.\ell}\left(\log\sigma_{k\ell}^2 + \frac{u_{i\ell}^{\widetilde{\mathbf{w}}} - 2\mu_{k\ell}x_{i\ell}^{\widetilde{\mathbf{w}}} + \mu_{k\ell}^2}{\sigma_{k\ell}^2}\right)\right)$

step 2. $\pi_k = \frac{\widetilde{z}_{.k}}{n}$, $\mu_{k\ell} = \frac{\sum_i \widetilde{z}_{ik}x_{i\ell}^{\widetilde{\mathbf{w}}}}{\widetilde{z}_{.k}}$, $\sigma_{k\ell}^2 = \frac{\sum_i \widetilde{z}_{ik}u_{i\ell}^{\widetilde{\mathbf{w}}}}{\widetilde{z}_{.k}} - \mu_{k\ell}^2$

until convergence

$x_{kj}^{\widetilde{\mathbf{z}}} = \frac{1}{\widetilde{z}_{.k}}\sum_i \widetilde{z}_{ik}x_{ij}$, $v_{kj}^{\widetilde{\mathbf{z}}} = \frac{1}{\widetilde{z}_{.k}}\sum_i \widetilde{z}_{ik}x_{ij}^2$

repeat

step 3. $\widetilde{\mathrm{w}}_{j\ell} \propto \rho_\ell \exp\left(-\frac{1}{2}\sum_k \widetilde{z}_{.k}\left(\log\sigma_{k\ell}^2 + \frac{v_{kj}^{\widetilde{\mathbf{z}}} - 2\mu_{k\ell}x_{kj}^{\widetilde{\mathbf{z}}} + \mu_{k\ell}^2}{\sigma_{k\ell}^2}\right)\right)$

step 4. $\rho_\ell = \frac{\widetilde{\mathrm{w}}_{.\ell}}{d}$, $\mu_{k\ell} = \frac{\sum_j \widetilde{\mathrm{w}}_{j\ell}x_{kj}^{\widetilde{\mathbf{z}}}}{\widetilde{\mathrm{w}}_{.\ell}}$, $\sigma_{k\ell}^2 = \frac{\sum_j \widetilde{\mathrm{w}}_{j\ell}v_{kj}^{\widetilde{\mathbf{z}}}}{\widetilde{\mathrm{w}}_{.\ell}} - \mu_{k\ell}^2$

until convergence

until convergence

return π, ρ, α

5.2.3. *Parsimonious Gaussian latent block models*

As in section 1.7.2 of Chapter 1, a parsimonious model can be defined by imposing constraints on the variances: we obtain the $[\sigma]$ model when the variances depend neither on the row cluster nor on the column cluster, the $[\sigma_k]$ model when the variances depend only on the row cluster, and the $[\sigma^j]$ model when the variances depend only on the column cluster.

This parsimonious parameterization allows us to study the relationship between CROEUC algorithm described in section 5.1 and the latent block model. Indeed, in the simplest case, in the $[\sigma]$ model, given identical proportions ($\pi_k = 1/g$ and $\rho_\ell = 1/m$), the complete-data log-likelihood

takes the form

$$\mathrm{L_C}(\mathbf{z}, \mathbf{w}, \boldsymbol{\alpha}) = -\frac{nd}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i,j,k,\ell} z_{ik}w_{j\ell}(x_{ij} - \mu_{k\ell})^2 - n\log g - d\log m$$

and it is easy to see that maximizing $\mathrm{L_C}$ is equivalent to minimizing $\mathrm{W}(\mathbf{z}, \mathbf{w})$ where

$$\mathrm{W}(\mathbf{z}, \mathbf{w}) = \sum_{i,j,k,\ell} z_{ik}w_{j\ell}(x_{ij} - x_{k\ell}^{\mathbf{zw}})^2$$

is the criterion maximized by the CROEUC algorithm.

Algorithm 5.4 Gaussian LBCEM

input: $\mathbf{x}$, g, m
initialization: $\mathbf{z}$, $\mathbf{w}$, $\pi_k = \frac{z_{\cdot k}}{n}$ $\rho_\ell = \frac{w_{\cdot \ell}}{d}$, $\mu_{k\ell} = \frac{x_{k\ell}^{\mathbf{zw}}}{z_{\cdot k}w_{\cdot \ell}}$. $\sigma_{k\ell}^2 = \frac{\sum_{ij} z_{ik}w_{j\ell}x_{ij}^2}{z_{\cdot k}w_{\cdot \ell}} - \mu_{k\ell}^2$
repeat
  $x_{i\ell}^{\mathbf{w}} = \frac{1}{w_{\cdot \ell}}\sum_j w_{j\ell}x_{ij}$, $u_{i\ell}^{\mathbf{w}} = \frac{1}{w_{\cdot \ell}}\sum_j w_{j\ell}x_{ij}^2$
  repeat
    step 1. $z_i = \arg\max_k \log\pi_k - \frac{1}{2}\sum_\ell w_{\cdot \ell}\left(\log\sigma_{k\ell}^2 + \frac{u_{i\ell}^{\mathbf{w}} - 2\mu_{k\ell}x_{i\ell}^{\mathbf{w}} + \mu_{k\ell}^2}{\sigma_{k\ell}^2}\right)$
    step 2. $\pi_k = \frac{z_{\cdot k}}{n}$, $\mu_{k\ell} = \frac{\sum_i z_{ik}x_{i\ell}^{\mathbf{w}}}{z_{\cdot k}}$, $\sigma_{k\ell}^2 = \frac{\sum_i z_{ik}u_{i\ell}^{\mathbf{w}}}{z_{\cdot k}} - \mu_{k\ell}^2$
  until convergence
  $x_{kj}^{\mathbf{z}} = \frac{1}{z_{\cdot k}}\sum_i z_{ik}x_{ij}$, $v_{kj}^{\mathbf{z}} = \frac{1}{z_{\cdot k}}\sum_i z_{ik}x_{ij}^2$
  repeat
    step 3. $w_j = \arg\max_\ell \log\rho_\ell - \frac{1}{2}\sum_k z_{\cdot k}\left(\log\sigma_{k\ell}^2 + \frac{v_{kj}^{\mathbf{z}} - 2\mu_{k\ell}x_{kj}^{\mathbf{z}} + \mu_{k\ell}^2}{\sigma_{k\ell}^2}\right)$
    step 4. $\rho_\ell = \frac{w_{\cdot \ell}}{d}$, $\mu_{k\ell} = \frac{\sum_j w_{j\ell}x_{kj}^{\mathbf{z}}}{w_{\cdot \ell}}$, $\sigma_{k\ell}^2 = \frac{\sum_j w_{j\ell}v_{kj}^{\mathbf{z}}}{w_{\cdot \ell}} - \mu_{k\ell}^2$
  until convergence
until convergence
return $\mathbf{z}$, $\mathbf{w}$, $\boldsymbol{\pi}$, $\boldsymbol{\rho}$, $\boldsymbol{\alpha}$

We now establish connections between LBCEM and CROEUC. To this end, it suffices to remark that in step 1 of LBCEM, we have

$$z_i = \arg\max_k \log \pi_k - \frac{1}{2} \sum_\ell w_{\cdot\ell} \left(\log \sigma_{k\ell}^2 + \frac{u_{i\ell}^{\mathbf{w}} - 2\mu_{k\ell} x_{i\ell}^{\mathbf{w}} + \mu_{k\ell}^2}{\sigma_{k\ell}^2} \right). \quad [5.6]$$

For the $[\sigma]$ model, this leads to

$$z_i = \arg\min_k \sum_\ell w_{\cdot\ell} \left(u_{i\ell}^{\mathbf{w}} - 2\mu_{k\ell} x_{i\ell}^{\mathbf{w}} + \mu_{k\ell}^2 \right)$$

or

$$z_i = \arg\min_k \sum_\ell \left(w_{\cdot\ell} (u_{i\ell}^{\mathbf{w}} - (x_{i\ell}^{\mathbf{w}})^2) + w_{\cdot\ell} (x_{i\ell}^{\mathbf{w}} - \mu_{k\ell})^2 \right)$$

and finally, since the first term does not depend on k, z_{ik} becomes

$$z_i = \arg\min_k \sum_\ell w_{\cdot\ell} (x_{i\ell}^{\mathbf{w}} - \mu_{k\ell})^2.$$

In the same way, we can prove that in step 3 of LBCEM, we have

$$w_j = \arg\min_\ell \sum_k z_{\cdot k} (x_{kj}^{\mathbf{z}} - \mu_{k\ell})^2.$$

Hence, it clearly appears that the CROEUC algorithm is just a particular version of the Gaussian LBCEM algorithm for the parsimonious model $[\sigma^2]$ assuming equal proportions.

5.3. Illustrative example

To illustrate the LBVEM algorithm with the $[\sigma_{k\ell}]$ model, we use the Amiard fishes data set presented in [CAI 76, p. 277].

Although the size of this data set is small, it is interesting to note that the number of variables is large compared to the small number of individuals. For the sake of brevity, we will only note that the data consist of a set of 24 fish placed in a radioactive environment. The first nine variables (ey: radioactivity of eyes, gi: radioactivity of gills, ca: radioactivity of cappings, fi: radioactivity of fins, le: radioactivity of lever, gt: radioactivity of the gastrointestinal trac, ki: radioactivity of kidney, sc: radioactivity of scales, mu: radioactivity of muscles) correspond to the level of radioactivity in different organs and the remaining variables (wgt: weight, l: length, sl: standard length, whe: width of head, w: width, wsn: width of snout, dey: diameter of eyes) correspond to different sizes. The 17th fish died during the experiment. Tables 5.2 and 5.3 present the standardized data matrix and the correlation data matrix.

We run the LBVEM algorithm with $(g, m) = (5, 3)$ that seems most relevant, we retain the best co-clustering and we report the clusters in Table 5.4 and on the two planes obtained by PCA and depicted in Figures 5.2 and 5.3. In Table 5.5, we present the mean and variance of each block allowing us to evaluate their degree of homogeneity. We can observe, for instance, the opposition between row clusters 3 and 4 that are characterized by column clusters 1 (level radioactivity) and 2 (size). Row cluster 3 has the smaller mean value for the level of radioactivity in different organs except *rki* and a higher mean for different sizes; while row cluster 4 presents the higher values for level of radioactivity (column cluster 1) and smaller mean for different sizes (column cluster 2). Note that this opposition is confirmed in Figure 5.2. Furthermore, we observe the opposition between row clusters 1 and 2 due to column cluster 1 (size). Row cluster 1 presents the smallest mean for column cluster 1 while row cluster 2 presents the higher mean for the level of radioactivity. Row cluster 5 does not seem relevant, its averages are fairly close to 0 and also located near the origin (Figure 5.2). Finally, column cluster 3 (radioactivity of kidney) seems relevant mainly for row cluster 4.

Ind.	rey	rgi	rca	rfi	rle	rgt	rki	rsc	rmu	wgt	l	sl	whe	w	wsn	dey
1	-0.724	-0.776	-0.598	-0.650	-1.228	-0.794	2.167	-0.737	-0.842	1.894	1.314	1.677	2.335	1.677	1.748	1.308
2	-0.857	-1.202	-0.950	-1.100	0.108	-0.651	0.286	-0.941	-0.470	1.515	1.649	1.740	1.294	1.012	0.960	0.271
3	-1.257	-1.124	-0.516	-0.785	-0.985	-0.346	-0.027	-0.836	-0.470	1.780	1.649	1.740	1.294	1.234	1.354	1.308
4	-1.124	-0.679	-0.936	-1.111	-1.167	0.110	0.286	-0.983	-0.470	1.932	1.928	1.804	1.918	1.899	0.565	1.308
5	-0.990	-0.892	-0.571	-0.729	-0.803	0.029	-1.595	-0.254	-0.842	-0.952	-1.256	-1.383	-1.203	-0.540	-1.799	-0.767
6	-0.990	-1.144	-0.733	-0.594	-0.620	-0.643	-0.341	-0.685	-0.842	-0.876	-0.697	-0.682	-0.995	-0.983	-1.011	-0.767
7	-1.124	-1.124	-0.990	-0.875	-0.681	-0.701	-1.595	-0.642	-0.842	-0.876	-0.809	-0.937	-0.579	-0.761	-1.011	-0.767
8	-0.591	-0.504	-0.855	-0.785	-0.438	-0.678	0.286	-0.742	0.275	-1.331	-0.809	-0.363	-0.787	-1.870	-1.405	-1.805
9	-0.324	-0.485	-0.611	-0.111	0.898	-0.346	-0.027	-0.609	-0.097	-0.383	-0.474	-0.427	-0.579	-0.096	-0.617	0.271
10	0.741	0.871	0.079	-0.212	1.323	-0.203	-0.027	-0.306	0.648	-0.117	0.532	0.529	0.461	-0.318	-0.617	-0.767
11	-0.458	-0.272	-0.340	-0.302	-0.317	1.190	-0.027	-0.368	-0.470	-0.079	-0.306	-0.491	0.045	0.347	-0.617	1.308
12	-0.191	0.290	-0.449	-0.448	-0.378	0.106	-0.027	1.511	0.648	-0.383	-0.865	-0.809	-0.579	-0.096	-0.223	0.271
13	-0.191	0.716	-0.313	-0.336	0.412	0.932	-0.027	-0.410	2.511	-0.269	-0.083	-0.108	-0.163	-0.096	1.748	0.271
14	1.008	-0.253	-0.395	-0.369	1.323	0.685	-0.027	-0.477	-0.470	-1.142	-1.479	-1.510	-1.412	-0.983	-0.617	-0.767
15	-0.324	-0.388	-0.611	-0.459	-0.438	0.140	-0.027	-0.505	0.275	0.148	0.253	0.274	-0.371	-0.096	0.960	0.271
16	-0.191	0.019	-0.571	-0.617	0.230	-0.643	-0.027	-0.235	1.021	0.186	1.091	-0.044	0.670	0.125	1.354	0.271
18	2.207	2.304	2.043	1.676	0.533	-0.674	1.227	0.773	-0.097	-0.383	-0.529	-0.427	-0.371	-0.761	-0.223	-0.767
19	0.875	1.103	1.474	1.721	-0.135	2.325	-1.281	1.383	-0.470	-0.724	-0.865	-0.682	-0.995	-0.983	-0.617	-0.767
20	2.074	1.743	1.339	2.080	2.780	-0.666	-1.281	2.419	2.884	-1.256	-1.144	-1.064	-0.787	-1.427	-0.617	-1.805
21	-0.058	0.425	0.134	0.361	-0.256	-0.705	-0.654	-0.666	-0.470	0.945	0.755	0.912	0.878	1.234	0.565	1.308
22	0.875	1.065	1.989	1.316	-0.924	-0.693	0.600	1.856	-0.097	0.035	-0.027	0.338	-0.163	1.012	0.171	-0.767
23	1.141	1.103	1.651	1.608	1.444	2.892	2.481	1.232	-0.470	-0.003	0.197	-0.172	-0.163	-0.096	0.171	0.271
24	0.475	-0.795	0.730	0.721	-0.681	-0.666	-0.341	0.224	-0.842	0.338	-0.027	0.083	0.253	0.569	-0.223	1.308

Table 5.2. *Standardized Amiard fishes data set*

	rey	rgi	rca	rfi	rle	rgt	rki	rsc	rmu	wgt	l	sl	whe	w	wsn	dey
rey	1.0	0.9	0.9	0.9	0.7	0.2	0.2	0.7	0.4	−0.4	−0.4	−0.4	−0.3	−0.4	−0.1	−0.4
rgi	0.9	1.0	0.8	0.8	0.6	0.3	0.2	0.7	0.5	−0.3	−0.3	−0.3	−0.2	−0.3	−0.0	−0.4
rca	0.9	0.8	1.0	1.0	0.4	0.3	0.2	0.8	0.1	−0.2	−0.2	−0.2	−0.2	−0.1	−0.1	−0.3
rfi	0.9	0.8	1.0	1.0	0.5	0.3	0.1	0.8	0.2	−0.3	−0.3	−0.3	−0.3	−0.3	−0.2	−0.3
rle	0.7	0.6	0.4	0.5	1.0	0.2	0.0	0.4	0.6	−0.4	−0.3	−0.4	−0.4	−0.5	−0.2	−0.4
rgt	0.2	0.3	0.3	0.3	0.2	1.0	0.2	0.3	−0.0	−0.2	−0.2	−0.2	−0.3	−0.1	−0.0	0.1
rki	0.2	0.2	0.2	0.1	0.0	0.2	1.0	−0.0	−0.1	0.4	0.4	0.4	0.4	0.4	0.5	0.3
rsc	0.7	0.7	0.8	0.8	0.4	0.3	−0.0	1.0	0.4	−0.4	−0.4	−0.4	−0.4	−0.3	−0.2	−0.4
rmu	0.4	0.5	0.1	0.2	0.6	−0.0	−0.1	0.4	1.0	−0.3	−0.1	−0.2	−0.2	−0.3	0.2	−0.3
wgt	−0.4	−0.3	−0.2	−0.3	−0.4	−0.2	0.4	−0.4	−0.3	1.0	0.9	0.9	0.9	0.9	0.7	0.8
l	−0.4	−0.3	−0.2	−0.3	−0.3	−0.2	0.4	−0.4	−0.1	0.9	1.0	1.0	0.9	0.8	0.8	0.7
sl	−0.4	−0.3	−0.2	−0.3	−0.4	−0.2	0.4	−0.4	−0.2	0.9	1.0	1.0	0.9	0.8	0.7	0.6
whe	−0.3	−0.2	−0.2	−0.3	−0.4	−0.3	0.4	−0.4	−0.2	0.9	0.9	0.9	1.0	0.9	0.7	0.7
w	−0.4	−0.3	−0.1	−0.3	−0.5	−0.1	0.4	−0.3	−0.3	0.9	0.8	0.8	0.9	1.0	0.7	0.8
wsn	−0.1	−0.0	−0.1	−0.2	−0.2	−0.0	0.5	−0.2	0.2	0.7	0.8	0.7	0.7	0.7	1.0	0.6
dey	−0.4	−0.4	−0.3	−0.3	−0.4	0.1	0.3	−0.4	−0.3	0.8	0.7	0.6	0.7	0.8	0.6	1.0

Table 5.3. *Correlation matrix*

Row clusters	Column clusters
cluster 1 (4 obs.) : 5 6 7 8	cluster 1 (7 obs.) : rey rgi rca rfi rle rgt rsc rmu
cluster 2 (4 obs.) : 18 22 23 24	cluster 2 (8 obs.) : wgt l sl whe w wsn dey
cluster 3 (5 obs.) : 1 2 3 4 21	cluster 3 (1 obs.) : rki
cluster 4 (2 obs.) : 19 20	
cluster 5 (8 obs.) : 9 10 11 12 13 14 15 16	

Table 5.4. *Row and column clusters*

$\mu_{k\ell}$	1	2	3
1	-0.71	-1.00	-0.81
2	1.43	-0.98	-1.28
3	-0.66	1.39	0.41
4	0.75	-0.01	0.99
5	0.03	-0.12	-0.034

$\sigma^2_{k\ell}$	1	2	3
1	0.09	0.14	0.66
2	1.10	0.11	0.00
3	0.19	0.21	0.89
4	1.14	0.22	1.05
5	0.43	0.44	0.00

Table 5.5. *Means and variances of each block*

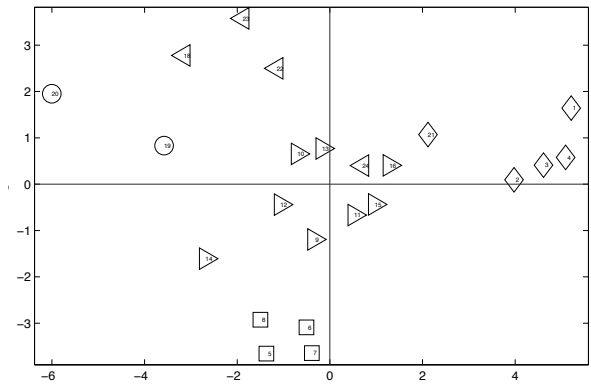

Figure 5.2. *Projection of the columns into the factorial plane spawned by the first and second axes that account for 71.07% of variance*

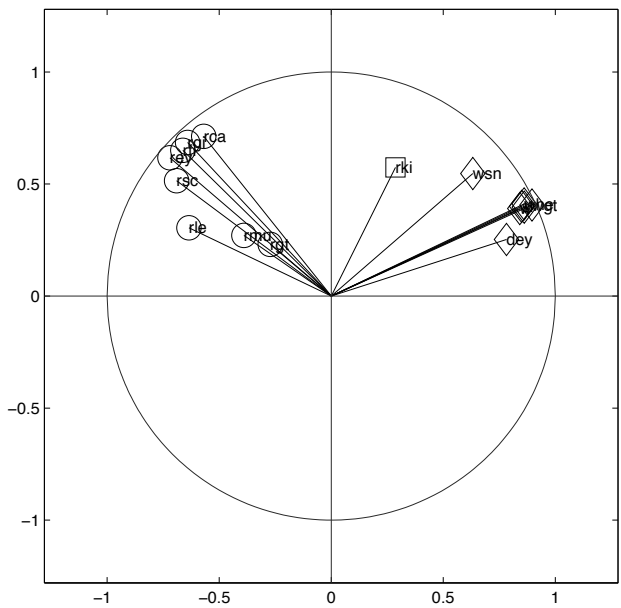

Figure 5.3. *Projection of the columns into the factorial plane spawned by the first and second axes that account for 71.07% of variance*

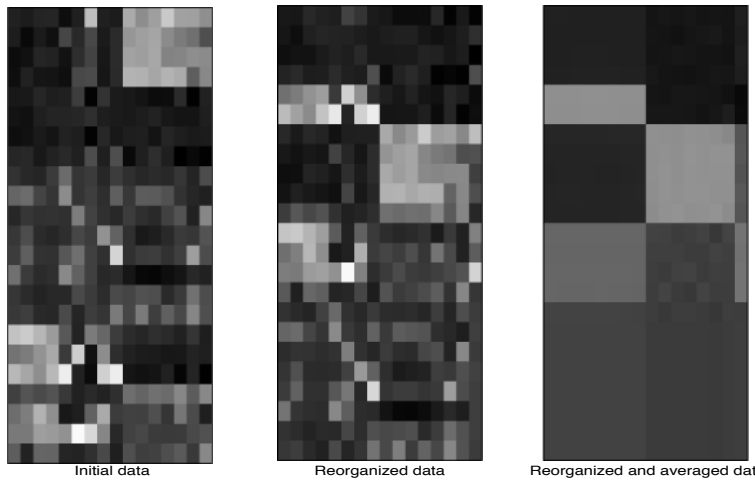

Figure 5.4. *Visualization of initial data and results after co-clustering*

5.4. Gaussian block mixture model

The Gaussian latent block model, being symmetric in both rows and columns, can appear less adapted to continuous data. In this case, the data matrix presents a dissymmetry: rows and columns that correspond to entities of different types will not be handled in the same way, contrary to what we have done in the two previous chapters for binary and contingency tables. Let us recall that we encounter the same problem with PCA where the individuals and the variables are not treated in a symmetrical way, which is not the case of *correspondence analysis* that treats the rows and the columns of a contingency table in the same way.

To overcome this difficulty, another model based on the classical Gaussian mixture model can be proposed. In this section, we describe this model and establish some connections with the latent block model.

5.4.1. *The model*

Hereafter, we use a classical mixture model in which the partition w of the variables is considered as a parameter of the model [NAD 10]. The pdf is therefore

$$f(\mathbf{x}_i;\boldsymbol{\theta}) = \sum_k \pi_k f(\mathbf{x}_i;\mathbf{w},\boldsymbol{\alpha}),$$

with $f(\mathbf{x}_i;\mathbf{w},\boldsymbol{\alpha}) = \prod_{j,\ell}\left(\frac{1}{\sqrt{2\pi\sigma_{k\ell}^2}}e^{-\frac{1}{2\sigma_{k\ell}^2}(x_{ij}-a_{k\ell})^2}\right)^{w_{j\ell}}$. The unknown parameter $\boldsymbol{\theta}$ is now formed by $\boldsymbol{\pi}$, w and $\boldsymbol{\alpha}$ where $\boldsymbol{\alpha} = (\mathbf{a},\Sigma)$ with a and Σ being $g \times m$ matrices representing the means and the variances of blocks

$$\mathbf{a} = \begin{pmatrix} a_{11} & \dots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{g1} & \dots & a_{gm} \end{pmatrix}, \Sigma = \begin{pmatrix} \sigma_{11}^2 & \dots & \sigma_{1m}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{g1}^2 & \dots & \sigma_{gm}^2 \end{pmatrix},$$

or

$$\boldsymbol{\alpha} = \begin{pmatrix} (a_{11},\sigma_{11}^2) & \dots & (a_{1m},\sigma_{1m}^2) \\ \vdots & \ddots & \vdots \\ (a_{g1},\sigma_{g1}^2) & \dots & (a_{gm},\sigma_{gm}^2) \end{pmatrix}.$$

This model can be viewed as a Gaussian mixture model with constraints on the g mean vectors and g variance matrices. For each component k, the $(d \times 1)$ mean vector $\mathbf{a}_k$ takes this form

$$(a_{k1},\dots,a_{k1},a_{k2},\dots,a_{k2},\dots,a_{km},\dots,a_{km})^T,$$

where each $a_{k\ell}$ is repeated w_ℓ times. In the same manner, the variance matrix Σ_k is a diagonal $(d \times d)$ matrix defined by

$$Diag(\sigma_{k1}^2,\dots,\sigma_{k1}^2,\sigma_{k2}^2,\dots,\sigma_{k2}^2,\dots,\sigma_{km}^2,\dots,\sigma_{km}^2),$$

where each variance $\sigma^2_{k\ell}$ is repeated w_ℓ times. When for each component k the variances are assumed to be equal to σ^2_k, Σ_k becomes $\sigma^2_k I$. This is a parsimonious model, as opposed to a spherical Gaussian mixture model. The number of parameters is equal to $g + 2(g * m)$ instead of $g + 2(g * d)$. Hence, it is more adapted when n is much smaller than d, a classical situation in bioinformatics.

5.4.2. GBEM *algorithm*

Setting this model under the maximum likelihood approach, the EM algorithm can be used to estimate the parameters. The complete-data log-likelihood

$$f(x; \boldsymbol{\theta}, \mathbf{z}) = \sum_{i,k} z_{ik} \log \left(\pi_k f(\mathbf{x}_i; \mathbf{w}, \boldsymbol{\alpha})\right)$$

takes, up to the constant $-\frac{nd}{2} \log 2\pi$, the following form

$$\mathrm{L_C}(\mathbf{z}; \boldsymbol{\theta}) = \sum_k z_{.k} \log \pi_k - \frac{1}{2} \sum_{i,j,k,\ell} z_{ik} w_{j\ell} \left(\log \sigma^2_{k\ell} + \frac{(x_{ij} - a_{k\ell})^2}{\sigma^2_{k\ell}} \right).$$

Using the interpretation of Hathaway [HAT 86] described in section 1.4.5 of Chapter 1, the EM algorithm can be seen as the alternating optimization algorithm of the fuzzy criterion

$$\mathrm{F_C}(\widetilde{\mathbf{z}}, \boldsymbol{\theta}) = \mathrm{L_C}(\boldsymbol{\theta}, \widetilde{\mathbf{z}}) + \mathrm{H}(\widetilde{\mathbf{z}}),$$

where $\mathrm{H}(\widetilde{\mathbf{z}}) = -\sum_{i,k} \widetilde{z}_{ik} \log \widetilde{z}_{ik}$ is the entropy of the distribution $\widetilde{\mathbf{z}}$. Therefore, the EM algorithm, denoted by Gaussian block EM algorithm (GBEM), alternates the following steps.

– ***E** step:* the maximization for fixed $\boldsymbol{\theta}$ reduces to the computation of the conditional probabilities. Each probability

$\widetilde{z}_{ik}$ is proportional to $\pi_k f(\mathbf{x}_i; \mathbf{w}, \boldsymbol{\alpha})$ whose logarithm takes the form

$$\log \pi_k - \frac{1}{2} \sum_{\ell} \left(w_\ell \log \sigma_{k\ell}^2 + \frac{(e_{i\ell}^{\mathbf{w}} + w_\ell (x_{i\ell}^{\mathbf{w}} - a_{k\ell})^2)}{\sigma_{k\ell}^2} \right),$$

with $x_{i\ell}^{\mathbf{w}} = \frac{\sum_j w_{j\ell} x_{ij}}{w_\ell}$ and $e_{i\ell}^{\mathbf{w}} = \sum_j w_{j\ell} (x_{ij} - x_{i\ell}^{\mathbf{w}})^2$.

– ***M step:*** the maximization in $\boldsymbol{\theta}$ of $\mathrm{L_C}(\boldsymbol{\theta}, \widetilde{\mathbf{z}})$ is not straightforward. We can use the generalized EM algorithm (GEM) for which the M step requires $\boldsymbol{\theta}$ to be chosen such that $\mathrm{L_C}$ is increased rather than maximized over all $\boldsymbol{\theta}$. The maximization of $\sum_k \widetilde{z}_{\cdot k} \log \pi_k$ leads to $\pi_k = \frac{\widetilde{z}_{\cdot k}}{n}$ and to decrease

$$h(\mathbf{w}, \boldsymbol{\alpha}) = \sum_{i,j,k,\ell} \widetilde{z}_{ik} w_{j\ell} \left(\log \sigma_{k\ell}^2 + \frac{(x_{ij} - a_{k\ell})^2}{\sigma_{k\ell}^2} \right)$$

the following alternated minimizations can be used:

1) Computation of $\mathbf{w}$ given $\boldsymbol{\alpha}$: this step consists of minimizing h with respect to $\mathbf{w}$. As the expression of $h(\mathbf{w}, \boldsymbol{\alpha})$ can be written as $\sum_{j,\ell} w_{j\ell} t_{j\ell}$ where

$$t_{j\ell} = \sum_k \left(\widetilde{z}_{\cdot k} \log \sigma_{k\ell}^2 + \frac{1}{\sigma_{k\ell}^2} \left(f_{kj}^{\widetilde{\mathbf{z}}} + \widetilde{z}_{\cdot k} (x_{kj}^{\widetilde{\mathbf{z}}} - a_{k\ell})^2 \right) \right),$$

with $x_{kj}^{\widetilde{\mathbf{z}}} = \frac{\sum_i \widetilde{z}_{ik} x_{ij}}{\widetilde{z}_{\cdot k}}$ and $f_{kj}^{\widetilde{\mathbf{z}}} = \sum_i \widetilde{z}_{ik} (x_{ij} - x_{kj}^{\widetilde{\mathbf{z}}})^2$, we obtain $w_j = \arg\min_{\ell} t_{j\ell}$.

2) Computation of $\boldsymbol{\alpha}$ given $\mathbf{w}$: this step consists of minimizing h with respect to $\boldsymbol{\alpha}$ given $\mathbf{w}$. For all k and ℓ, the expression to minimize is

$$\widetilde{z}_{\cdot k} w_\ell \log \sigma_{k\ell}^2 + \sum_{i,j} \widetilde{z}_{ik} w_{j\ell} \frac{(x_{ij} - a_{k\ell})^2}{\sigma_{k\ell}^2},$$

which leads to

$$a_{k\ell} = \frac{\sum_{ij} \widetilde{z}_{ik} w_{j\ell} x_{ij}}{\widetilde{z}_{\cdot k} w_{\cdot \ell}} \quad \text{and} \quad \sigma_{k\ell}^2 = \frac{\sum_{i,j} \widetilde{z}_{ik} w_{j\ell} \left(x_{ij} - a_{k\ell} \right)^2}{\widetilde{z}_{\cdot k} w_\ell}.$$

Note that in the M step, using the terms $x_{kj}^{\widetilde{\mathbf{z}}}$ and $f_{kj}^{\widetilde{\mathbf{z}}}$, it is easy to show that the mean and the variance of each block

take, respectively, the following forms

$$a_{k\ell} = \frac{\sum_j w_{j\ell} x_{kj}^{\widetilde{\mathbf{z}}}}{w_\ell} \quad \text{and} \quad \sigma_{k\ell}^2 = \frac{\sum_j w_{j\ell} \left(f_{kj}^{\widetilde{\mathbf{z}}} + \widetilde{z}_{.k} (x_{kj}^{\widetilde{\mathbf{z}}} - a_{k\ell})^2 \right)}{\widetilde{z}_{.k} w_\ell}, \tag{5.7}$$

and computational shortcuts can be performed on a reduced matrix using sufficient statistics $x_{kj}^{\widetilde{\mathbf{z}}}$ and $f_{kj}^{\widetilde{\mathbf{z}}}$. The different steps of GBEM are summarized in algorithm 5.5.

Algorithm 5.5 GBEM

input: $\mathbf{x}$, g, m

initialization: $\mathbf{z}$, $\mathbf{w}$, $\pi_k = \frac{z_{.k}}{n}$, $a_{k\ell} = \frac{\sum_{ij} z_{ik} w_{j\ell} x_{ij}}{z_{.k} w_{.\ell}}$, $\sigma_{k\ell}^2 = \frac{\sum_{i,j} z_{ik} w_{j\ell} (x_{ij} - a_{k\ell})^2}{z_{.k} w_\ell}$

repeat

E-step: $x_{i\ell}^{\mathbf{w}} = \frac{\sum_j w_{j\ell} x_{ij}}{w_\ell}$, $e_{i\ell}^{\mathbf{w}} = \sum_j w_{j\ell} (x_{ij} - x_{i\ell}^{\mathbf{w}})^2$,

$\widetilde{z}_{ik} \propto \log \pi_k - \frac{1}{2} \sum_\ell \left(w_\ell \log \sigma_{k\ell}^2 + \frac{(e_{i\ell}^{\mathbf{w}} + w_\ell (x_{i\ell}^{\mathbf{w}} - a_{k\ell})^2)}{\sigma_{k\ell}^2} \right)$

M-step: $\pi_k = \frac{\widetilde{z}_{.k}}{n}$, $x_{kj}^{\widetilde{\mathbf{z}}} = \frac{\sum_i \widetilde{z}_{ik} x_{ij}}{\widetilde{z}_{.k}}$, $f_{kj}^{\widetilde{\mathbf{z}}} = \sum_i \widetilde{z}_{ik} (x_{ij} - x_{kj}^{\widetilde{\mathbf{z}}})^2$

repeat

Step a: $w_j = \arg\min_\ell \sum_k \left(\widetilde{z}_{.k} \log \sigma_{k\ell}^2 + \frac{1}{\sigma_{k\ell}^2} \left(f_{kj}^{\widetilde{\mathbf{z}}} + \widetilde{z}_{.k} (x_{kj}^{\widetilde{\mathbf{z}}} - a_{k\ell})^2 \right) \right)$

Step b: $a_{k\ell} = \frac{\sum_j w_{j\ell} x_{kj}^{\widetilde{\mathbf{z}}}}{w_\ell}$, $\sigma_{k\ell}^2 = \frac{\sum_j w_{j\ell} \left(f_{kj}^{\widetilde{\mathbf{z}}} + \widetilde{z}_{.k} (x_{kj}^{\widetilde{\mathbf{z}}} - a_{k\ell})^2 \right)}{\widetilde{z}_{.k} w_\ell}$

until convergence

until convergence

return $\boldsymbol{\pi}$, $\boldsymbol{\alpha}$

Regarding the context of clustering with the maximum likelihood approach, after we estimate parameter $\boldsymbol{\theta}$, we can give a probabilistic clustering of the n individuals in terms of their fitted posterior probabilities of component membership $\widetilde{z}_{ik}$ obtained at the convergence of EM. Therefore, we can obtain a partition by using a classification step that assigns each individual to the component of the mixture for which it has the highest posterior probability of belonging. With the

optimal w partition, we therefore obtain a co-clustering where a partition of individuals is characterized by a partition of variables. The GBEM algorithm can be viewed as a soft algorithm to cluster the set of individuals and the set of variables simultaneously.

A hard version called classification GBEM can be performed by replacing the maximization of $\mathrm{L}(\boldsymbol{\theta})$ by the maximization of $\mathrm{L_C}(\mathbf{z}, \mathrm{w}; \boldsymbol{\theta})$. The main modifications concern the conditional maximization of complete-data log-likelihoods with respect to w given z and $\boldsymbol{\theta}$, and with respect to $\boldsymbol{\theta}$ given z and w. This leads us to convert the posterior probabilities z_{ik}s into a discrete classification

$$z_{ik} = 1 \quad \text{if} \quad k = \underset{k'=1,\ldots,g}{\arg\max}\ \widetilde{z}_{ik'} \quad \text{and} \quad z_{ik} = 0 \quad \text{otherwise}$$

in a C step before performing the M step based this time on the clusters. It can also be remarked that this algorithm is the Gaussian LBCEM algorithm precisely when the column proportions ρ_ℓ are equal to $1/m$.

5.5. Numerical experiments

In these first experiments, we consider the model where all blocks have the same variance and the proportions of clusters are equal. We have chosen this restriction in order to evaluate the different algorithms in the same condition. First, to demonstrate the advantage of GBEM, we compared its performances with classical EM on the diagonal Gaussian model ignoring the clustering of variables. Second, we evaluated GBEM when the number of columns was higher the number of rows.

5.5.1. GBEM *versus* CROEUC *and EM*

To illustrate the behavior of GBEM, we selected $1{,}000 \times 50$ data arising from 3×2 component mixture models corresponding to three degrees of overlap of the clusters: well separated, moderately separated and poorly separated. The concept of cluster separation for our model is difficult to visualize, but the degree of overlap can be measured by the true error rate, approximated by comparing the partitions simulated with those that we obtained by applying a classification step. From our numerical experiments, we present only three situations corresponding to three levels of overlap degrees: M1 for well-separated clusters (8.6%), M2 for moderately separated clusters (16%) and M3 for poorly separated clusters (24.8%). To compare two partitions $\mathbf{z}$ and $\mathbf{z}'$ with the same number of clusters, the error rate or the proportions of misclassified individuals is denoted by $\delta(\mathbf{z}, \mathbf{z}')$. It can be defined as follows: if C is the confusion matrix between the two partitions, relabel the components of the partition $\mathbf{z}'$ such that the trace of matrix C is maximal (to obtain this maximum value in our experiments, we enumerate all possible relabellings), then compute

$$\delta(\mathbf{z}, \mathbf{z}') = 1 - \frac{1}{n} \sum_{i,k} z_{ik} z'_{ik}.$$

In Table 5.6, we compared the performances of GBEM, EM and CROEUC by using $\delta(\mathbf{z}, \mathbf{z}')$ (in percent) and their execution times recorded from the same initial positions. It is clear that GBEM outperforms EM and CROEUC. On the other hand, GBEM is faster than EM, the rate timeEM/timeGBEM denoted by t_{EM}/t_{GBEM} is higher than 2. Different Monte Carlo simulations were performed confirming these remarks and also the superiority of GBEM as compared to EM and CROEUC.

Error (%)	Situation	GBEM	EM	CROEUC	$\frac{t_{EM}}{t_{\text{GBEM}}}$
$\delta(\mathbf{z}, \mathbf{z}')$	M1	8.5	8.6	8.5	2.01
	M2	16.0	21.6	18.1	2.66
	M3	19.6	35.6	24	2.16

Table 5.6. *Comparison between* GBEM*,* EM *and* CROEUC *($n \times d = 1{,}000 \times 50$)*

5.5.2. *Effect of the size of data*

Now, we illustrate the interest of GBEM when n is less than d, which is a crucial problem in bioinformatics. As the latent block model is parsimonious, it does not suffer from this problem and therefore offers a good alternative for clustering individuals. Table 5.7 presents the degree of overlap and the error rates $\delta(\mathbf{z}, \mathbf{z}')$ for different sizes of n. We note that when n is less than d, GBEM is always the best even though n approaches d and remains the best when n is greater than d.

n	20	30	40	400
degree of overlap (%)	5	13	15	14
$\delta(\mathbf{z}, \mathbf{z}')$ for GBEM	5	13	17	14
$\delta(\mathbf{z}, \mathbf{z}')$ for EM	35	26	30	19

Table 5.7. GBEM *versus EM when $n < d = 400$*

5.6. Conclusion

For continuous data, we have considered their co-clustering under metric and probabilistic approaches. In the latter approach, we have considered the Gaussian latent block model and the Gaussian block mixture model in which the partition of the set of variables is considered as a parameter of the model. We have presented different algorithms and showed their connections. In addition, this probabilistic approach, as for binary data and contingency

tables, allows us to give a probabilistic interpretation of an objective function commonly used in co-clustering. In practice, it has been demonstrated with a simple example that co-clustering is very complementary to other exploratory methods such as PCA. To combine co-clustering and visualization, Priam *et al*. [PRI 13] proposed a Gaussian topographic co-clustering model based on the latent block model. Furthermore, when the data matrix is positive, many alternative methods based on non-negative matrix factorization can be proposed [LON 05, DIN 06, YOO 10, LAB 11a] and are frequently evaluated in the document clustering field on contingency tables after converting original data into continuous data using different types of normalization such as tf-idf. Note that this normalization step is crucial for the quality of results and deserves more investigation.