Model-Based Co-clustering for Continuous Data

Mohamed Nadif
LIPADE, UFR Mathématiques-Informatique
Université Paris Descartes, 45, rue des Saints-Pères
75270 Paris, France
mohamed.nadif@parisdescartes.fr

Gérard Govaert
Heudiasyc, CNRS 6599
Université de Technologie de Compiègne
60280 Compiègne, France
gerard.govaert@utc.fr

***Abstract*—The co-clustering consists in reorganizing a data matrix into homogeneous blocks by considering simultaneously the sets of rows and columns. Setting this aim in model-based clustering, adapted block latent models were proposed for binary data and co-occurrence matrix. Regarding continuous data, the latent block model is not appropriated in many cases. As non-negative matrix factorization, it treats symmetrically the two sets, and the estimation of associated parameters requires a variational approximation. In this paper we focus on continuous data matrix without restriction to non negative matrix. We propose a parsimonious mixture model allowing to overcome the limits of the latent block model.**

I. Introduction

Let $\mathbf{x}$ be a data matrix defined on two sets I (rows, objects, observations, cases) and J (columns, variables, attributes), the co-clustering methods aim to reorganize $\mathbf{x}$ into homogeneous blocks by considering simultaneously I and J. Here, we restrict to co-clustering methods defined by partitions of I and J. The basic principle of these methods is to make permutations of rows and columns in order to show block structure on $I \times J$. Another advantage of co-clustering methods is that they reduce $\mathbf{x}$ into a simpler one having the same structure (e.g. a binary data $\mathbf{x}$ is summarized by a binary data). Moreover, far less computation is required than for processing the two sets separately and consequently these methods are of interest in data mining. In this context, the co-clustering has become an important challenge. For instance, in the text mining field, by exploiting the duality between rows (documents) and columns (words), a spectral block clustering method has been proposed in [6] and a co-clustering based on the mutual information in [7]. In the analysis of microarray data where data are often presented as matrices of expression levels of genes under different conditions, co-clustering of genes and conditions has permitted to overcome the problem of the choice of similarity on the two sets found in conventional clustering methods [2].

Different approaches are employed to treat the co-clustering problem. Among them probabilistic model-based clustering techniques have shown promising results in several situations. For instance, the co-clustering of binary and contingency data has been treated by using latent block Bernoulli and Poisson models [11], [12].

In this paper we focus on the co-clustering of data matrix consisting of objects in the rows and continuous variables in the columns. We set this problem in the model based clustering context. The latent block model can be extended by using Gaussian distributions but the symmetric treatment of objects and variables is often not adapted. The sets of objects and variables are not comparable. We encounter the same problem with *principal component analysis* where the objects and the variables are not treated in a symmetrical way which is not the case of *correspondence analysis* which treats the rows and the columns of a co-occurrence matrix in the same way. Note that in this case, the co-clustering can be formulated as a matrix approximation problem as in the case of binary data [15] or in the case of co-occurrence matrix [16]. Different techniques frequently used, are based on the non-negative factorization which treats symmetrically the objects and the variables.

The first contribution of this paper is the proposition of a new mixture model applied on data matrix not necessarily non-negative and where both sets I and J are not treated symmetrically. The second contribution is that this model, thanks to the classification maximum approach, allows to give an interpretation to a classical criterion. Further we can propose other criteria. The last contribution is that our proposed model is parsimonious and adapted to data matrix when $|I| < |J|$ ($|.|$ denotes the cardinality).

The rest of the paper is organized as follows. Section 2 is devoted to review the problem of co-clustering for binary, dyadic data matrix and continuous data. In Section 3, we describe the latent block model and a new parsimonious mixture models adapted to our co-clustering problem. In Section 4, we estimate the model parameters by a *Block* EM algorithm. In Section 5, we describe a clustering version. To achieve our aim we study in Section 6 the behavior of this algorithm.

Notation: A partition of the set of objects into g clusters is noted $\mathbf{z}$ and will be represented by the classification matrix $(z_{ik}; i = 1, \ldots, n; k = 1, \ldots, g)$ where $z_{ik} = 1$ if i belongs to the kth cluster and 0 otherwise. A similar notation will be used for a partition $\mathbf{w}$ of the set of variables into m clusters represented also by the classification matrix $(w_{j\ell}; j = 1, \ldots, d; \ell = 1 \ldots, m)$. We denote the cardinalities of the kth and ℓth clusters by $z_k = \sum_{i=1}^{n} z_{ik}$ and $w_\ell = \sum_{j=1}^{d} w_{j\ell}$. To simplify the notation, the sums and the products relating to rows, columns or clusters will be subscripted respectively by letters i, j, k or ℓ without indicating the limits of variation, which will be implicit. Finally we denote a random variable by an upper case letter (e.g., X_{ij}) and the state or value of a corresponding variable by the same letter, in lower case (e.g., x_{ij}).

II. Co-clustering algorithms

A. General criterion

For market basket data or document clustering when the values are binary, the co-clustering becomes a classical approach. The detection of homogeneous blocks in data matrix $\mathbf{x}$ can be reached by partitioning the rows into g clusters and the columns into m clusters. Let be the non-negative arbitrary matrices $\mathbf{r} = (r_{ik})_{n\times g}$, $\mathbf{c} = (r_{j\ell})_{d\times m}$ and $\mathbf{a} = (a_{k\ell})_{g\times m}$ designating respectively row and column memberships and cluster representation which can be viewed as a summary of $\mathbf{x}$. The problem is to look for these three matrices minimizing the total squared residue measure

$$W(\mathbf{r}, \mathbf{c}, \mathbf{a}) = ||\mathbf{x} - \mathbf{rac}^T||^2, \tag{1}$$

where $||.||$ denote Frobenius matrix norm and the superscript T denotes matrix transposition. The term $\mathbf{rac}^T$ characterizes the information of $\mathbf{x}$ that can be described by the cluster structures. Then the clustering problem can be formulated as a matrix approximation problem where the clustering aim is to minimize the approximation error between the original data $\mathbf{x}$ and the reconstructed matrix based on the cluster structures.

The approximation of $\mathbf{x}$ can be solved by an iterative alternating least-squares optimization procedure. The non-negative block value decomposition (NBVD) [16] offers a solution of this problem. Furthermore, when $\mathbf{a}$ is identity matrix, this leads to the cluster model described in [15] and [17]. Note that both approaches can also be used in the case of dyadic data matrix such as co-occurrence matrix or when the values of data are continuous and positives. With these approaches by assuming that $\mathbf{ra}$ is normalized to $\mathbf{rav}$, the cluster labels of the columns, are deduced by $\mathbf{v}^{-1}\mathbf{c}^T = (c_{ij})$; $w_{j\ell} = 1$ if $\ell = \text{argmax}_{\ell'=1,\ldots,m}\ c_{j\ell'}$ and $w_{j\ell} = 0$ otherwise. We can also deduce the label cluster rows by working on $\mathbf{x}^T$.

B. Co-clustering for binary data

By imposing some constraints on $\mathbf{r}$, $\mathbf{c}$ and $\mathbf{a}$, we can propose different criteria. For example, if $\mathbf{r}$ and $\mathbf{c}$ are two classification matrices noted $\mathbf{z}$ and $\mathbf{w}$ and $\mathbf{a}$ is a binary data matrix, we can directly treat the co-clustering problem by minimizing

$$||\mathbf{x} - \mathbf{zaw}^T||^2.$$

Li [15] has proposed an algorithm based on the use of the double $k$means principle. The principal steps are

1) Start from an initial position $(\mathbf{z}^{(0)}, \mathbf{w}^{(0)}, \mathbf{a}^{(0)})$.
2) Computation of $(\mathbf{z}^{(c+1)}, \mathbf{w}^{(c+1)}, \mathbf{a}^{(c+1)})$ starting from $(\mathbf{z}^{(c)}, \mathbf{w}^{(c)}, \mathbf{a}^{(c)})$
   a) Update $\mathbf{a}^{(c+\frac{1}{2})}$: $a_{k\ell}^{(c+\frac{1}{2})} = \sum_{i,j} \frac{z_{ik}^{(c)} w_{j\ell}^{(c)} x_{ij}}{z_k^{(c)} w_\ell^{(c)}}$
   b) Update $\mathbf{z}^{(c+1)}$, each i belongs to the kth cluster minimizing $\sum_{j,\ell} w_{j\ell}^{(c)} (x_{ij} - a_{k\ell}^{(c+\frac{1}{2})})^2$.
   c) Update $\mathbf{w}^{(c+1)}$, each j belongs to the ℓth cluster minimizing $\sum_{i,k} z_{ik}^{(c)} (x_{ij} - a_{k\ell}^{(c+\frac{1}{2})})^2$.
   d) Computation of $\mathbf{a}^{(c+1)}$ as in (a) step.
3) Iterate the steps 2 until the convergence.

Obviously the update of $\mathbf{a}$ can be performed before the update of $\mathbf{w}$. This strategy appears more profitable because more faster. Furthermore, it exists an another version [8] more adapted for large data; it will be described in the case of continuous data not necessarily non-negative.

C. Continuous data

When the data are continuous, the sum of squared Euclidean distances can also be used as a measure of the deviation between the data matrices $\mathbf{x}$ and $\mathbf{zaw}^T$.

$$||\mathbf{x} - \mathbf{zaw}^T||^2 = \sum_{k,\ell} \sum_{i|z_{ik}=1} \sum_{j|w_{j\ell}=1} (x_{ij} - a_{k\ell})^2, \tag{2}$$

Different algorithms have been proposed to minimize this criterion (see for instance, [1], [3]). These algorithms are equivalent and consist in using the principle of a double $k$means. Furthermore, we recommend another version called *Croeuc* [8] based on the use of reduced intermediate matrices noted $\mathbf{u} = (u_{i\ell})$ and $\mathbf{v} = (v_{kj})$ where $u_{i\ell} = \sum_{j|w_{j\ell}=1} x_{ij}/w_\ell$ and $v_{kj} = \sum_{i|z_{ik}=1} x_{ij}/z_k$. These matrices appear naturally in the alternated steps. Indeed, the minimization of W can be performed by the two following conditional criteria

$$W(\mathbf{z}, \mathbf{a}|\mathbf{w}) = \sum_k \sum_{i|z_{ik}=1} w_\ell (u_{i\ell} - a_{k\ell})^2$$

and

$$W(\mathbf{w}, \mathbf{a}|\mathbf{z}) = \sum_\ell \sum_{j|w_{j\ell}=1} z_k (v_{kj} - a_{k\ell})^2.$$

These minimizations can be performed by using the k-means algorithm and *Croeuc* alternates these minimizations. In the first one, k-means is applied on the $n \times m$ matrix $\mathbf{u}$ with the Euclidean distance and the mean values of block clusters. the second step is carried out by the application of k-means on the $g \times d$ matrix $\mathbf{v}$ with the Euclidean distance and the mean values of block clusters. One repeats these steps and, at the convergence, one obtains homogeneous blocks by reorganizing rows and columns according to the partitions $\mathbf{z}$ and $\mathbf{w}$. Hence, each block $\mathbf{x}_{k\ell}$ is characterized by $a_{k\ell}$.

In fact, most of the algorithmic work on this problem has been heuristic in nature. The algorithms previously described might suffer from several problems. First, we can observe that the criterion W does not depend either on proportions of row and columns clusters nor of homogeneity degrees of block clusters. We will see how we can embed the co-clustering problem in the mixture approach and how we can propose efficient solutions.

III. Mixture model approach

A. Finite mixture model

Finite mixture models underpin a variety of techniques in major areas of statistics including cluster analysis. With a mixture model-based approach clustering, it is assumed that the data to be clustered are generated by a mixture of underlying probability distributions in which each component represents a different cluster. Given observations $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$, let

$\varphi(\mathbf{x}_i;\boldsymbol{\alpha}_k)$ be the density of an observation $\mathbf{x}_i$ from the kth component, where the $\boldsymbol{\alpha}_k$'s are the corresponding parameters and let g be the number of components in the mixture. The probability density function is

$$f(\mathbf{x}_i;\boldsymbol{\theta}) = \sum_{k=1}^{g} \pi_k \varphi(\mathbf{x}_i;\boldsymbol{\alpha}_k),$$

where π_k is the probability that an observation belongs to the kth component and $\boldsymbol{\theta}$ is the vector of the unknown parameters $(\pi_1,\ldots,\pi_g;\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_g)$.

Mixture models [13] may be used in two different ways to obtain a partition of the initial data. The first, known as the maximum likelihood (ML) approach, estimates the parameters of the model and then determines the partition $\mathbf{z}$ by allocating each row to the class that maximizes the a posteriori probability using these estimated parameters. The second, the classification maximum likelihood (CML) approach which involves creating a partition of the sample such that each kth class is made to correspond to a sub-sample respecting the distribution $\varphi(\mathbf{x}_i;\boldsymbol{\alpha}_k)$. In the ML and CML approaches the commonly used algorithms are EM [5] and Classification EM (CEM) [4].

B. Latent block model

Note that the mixture density of the observed data $\mathbf{x}$ can be expressed as $f(\mathbf{x},\boldsymbol{\theta}) = \prod_i \sum_k \pi_k \varphi(\mathbf{x}_i;\boldsymbol{\alpha}_k)$. This probability density function can be written as (see for instance [9])

$$f(\mathbf{x},\boldsymbol{\theta}) = \sum_{\mathbf{z}\in Z} p(\mathbf{z};\boldsymbol{\theta}) f(\mathbf{x}|\mathbf{z};\boldsymbol{\theta}) \tag{3}$$

where Z denotes the set of all possible assignments of objects into g clusters,

$$p(\mathbf{z};\boldsymbol{\theta}) = \prod_{i,k} \pi_k^{z_{ik}} \text{ and } f(\mathbf{x}|\mathbf{z};\boldsymbol{\theta}) = \prod_{i,k} \varphi(\mathbf{x}_i;\boldsymbol{\alpha}_k)^{z_{ik}}.$$

In the context of co-clustering, the formulation (3) can be extended to propose a latent block model defined by the following probability density function [9]:

$$f(\mathbf{x},\boldsymbol{\theta}) = \sum_{(\mathbf{z},\mathbf{w})\in Z\times W} p(\mathbf{z};\boldsymbol{\theta}) p(\mathbf{w};\boldsymbol{\theta}) f(\mathbf{x}|\mathbf{z},\mathbf{w};\boldsymbol{\theta}) \tag{4}$$

where Z and W denote the sets of all possibles assignments $\mathbf{z}$ of objects and $\mathbf{w}$ of variables. In this model we also assume local independence i.e., the $n \times d$ random variables X_{ij} are assumed to be independent once $\mathbf{z}$ and $\mathbf{w}$ are fixed; we have

$$f(\mathbf{x}|\mathbf{z},\mathbf{w};\boldsymbol{\theta}) = \prod_{i,j,k,\ell} \varphi(\mathbf{x}_{ij};\boldsymbol{\alpha}_k)^{z_{ik}w_{j\ell}}$$

where $\varphi(.;\alpha_{k\ell})$ is a probability density function defined on the real set $\mathbb{R}$. This model allows to propose algorithms for co-clustering binary and contingency tables by considering respectively Bernoulli and Poisson latent block models (see for instance; [11] and [12]). From these works, setting the clustering problem under the CML approach, we can show that the co-clustering of co-occurrence matrix by block value decomposition [16] and the co-clustering of binary data by [15] are respectively associated to restricted Poisson and Bernoulli latent block models. The authors have proposed different variant algorithms of EM based respectively on the variational approximation of the likelihood and the complete data likelihood.

For continuous, this model can be easily used by considering a latent Gaussian block model and the associated algorithms can be performed. Note that it is easy to show that the minimization of (2) is associated to Latent block Gaussian model where the proportions of row clusters and column clusters are equal and in addition the variances of blocks are identical. This leads to note the following remarks 1) the characteristic of the latent block model is that the rows and the columns are treated symmetrically 2) the estimation of the parameters requires a variational approximation [10]. To overcome these difficulties, we propose, in the following section, a new model.

C. A Parsimonious mixture model for co-clustering

Hereafter, we propose to use the classical mixture model in which the partition $\mathbf{w}$ of the variables is considered as a parameter of the model. The pdf is then

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

with

$$\varphi(\mathbf{x}_i;\mathbf{w},\boldsymbol{\alpha}) = \prod_{j,\ell} \left(\frac{1}{\sqrt{2\pi\sigma_{k\ell}^2}} e^{-\frac{1}{2\sigma_{k\ell}^2}(x_{ij}-a_{k\ell})^2} \right)^{w_{j\ell}}.$$

The unknown parameter $\boldsymbol{\theta}$ is formed now by $\mathbf{w}$, $\boldsymbol{\alpha}$ and $\boldsymbol{\pi} = (\pi_1,\ldots,\pi_g)$. The parameter $\boldsymbol{\alpha} = (\mathbf{a},\Sigma)$ where $\mathbf{a}$ and Σ are $g \times m$ matrices representing the means and the variances of blocks

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

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 $(p\times 1)$ mean vector $\mathbf{a}_k$ takes this form

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

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

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

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

IV. Block EM algorithm

Setting our model under the Maximum Likelihood (ML) approach, we propose to use the EM algorithm to estimate the parameters. The log-likelihood of observed data is

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

and, the complete data log-likelihood $L_c(\mathbf{z};\boldsymbol{\theta})$ is $\sum_{i,k} z_{ik} \log\left(\pi_k \varphi_k(\mathbf{x}_i;\mathbf{w},\boldsymbol{\alpha})\right)$. It takes, up to the constant $-\frac{nd}{2}\log 2\pi$, the following form:

$$\sum_k z_k \log \pi_k - \frac{1}{2}\sum_{i,j,k,\ell} z_{ik} w_{j\ell}\left(\log \sigma_{k\ell}^2 + \frac{(x_{ij}-a_{k\ell})^2}{\sigma_{k\ell}^2}\right).$$

We can extend this complete data log-likelihood L_c, defined on a partition $\mathbf{z}$, to the fuzzy partition associated to $\mathbf{s} = (s_{ik}; i = 1,\ldots,n; k = 1,\ldots,g)$ the classification matrix defined by the conditional probabilities. The expression of $L_c(\mathbf{s};\boldsymbol{\theta})$ is equal to

$$\sum_k s_k \log \pi_k - \sum_{i,j,k,\ell} \frac{s_{ik} w_{j\ell}}{2}\left(\log \sigma_{k\ell}^2 + \frac{(x_{ij}-a_{k\ell})^2}{\sigma_{k\ell}^2}\right),$$

where $s_k = \sum_i s_{ik}$.

Starting from $\boldsymbol{\theta}^{(0)}$, the EM algorithm alternates the following steps.

A. Estimation step

This step reduces to the computation of the conditional probabilities. Each probability $s_{ik}^{(c)}$ is proportional to $\pi_k^{(c)}\varphi_k(\mathbf{x}_i;\mathbf{w}^{(c)},\boldsymbol{\alpha}^{(c)})$ where the logarithm takes this form

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

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

B. Maximization step

The maximization of $Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(c)})$ is not straightforward. We can use the Generalized EM algorithm (GEM) for which the M-step requires $\boldsymbol{\theta}^{(c+1)}$ to be chosen such that $Q(\boldsymbol{\theta}^{(c+1)},\boldsymbol{\theta}^{(c)}) \geq Q(\boldsymbol{\theta}^{(c)},\boldsymbol{\theta}^{(c)})$: that is, one chooses $\boldsymbol{\theta}^{(c+1)}$ to increase the function $Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(c)})$ rather than maximize it over all $\boldsymbol{\theta}$. Note that $Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(c)})$ is the fuzzy complete data log-likelihood

$$L_c(\mathbf{s}^{(c)};\boldsymbol{\theta}) = \sum_k s_k^{(c)} \log \pi_k - \frac{1}{2}H(\mathbf{w},\boldsymbol{\alpha}) \text{ with}$$

$$H(\mathbf{w},\boldsymbol{\alpha}) = \sum_{i,j,k,\ell} s_{ik}^{(c)} w_{j\ell}(\log \sigma_{k\ell}^2 + \frac{(x_{ij}-a_{k\ell})^2}{\sigma_{k\ell}^2}).$$

The maximization of $\sum_k s_k^{(c)} \log \pi_k$ leads to $\pi_k^{(c+1)} = \frac{s_k^{(c)}}{n}$ and to decrease $H(\mathbf{w},\boldsymbol{\alpha})$ we propose the following alternated minimizations.

1) Computation of $\mathbf{w}$ *given* $\boldsymbol{\alpha}$*:* This step consists in minimizing $H(\mathbf{w},\boldsymbol{\alpha})$ w.r. to $\mathbf{w}$. The expression of $H(\mathbf{w}^{(c+1)},\boldsymbol{\alpha})$ can be written as $\sum_{j,\ell} w_{j\ell}^{(c+1)} T_{j\ell}^{(c)}$ where

$$T_{j\ell}^{(c)} = \sum_k (s_k^{(c)} \log \sigma_{k\ell}^2 + \frac{1}{\sigma_{k\ell}^2}\left(f_{kj} + s_k (v_{kj}-a_{k\ell})^2\right))$$

with $v_{kj} = \frac{\sum_i s_{ik} x_{ij}}{s_k}$ and $f_{kj} = \sum_i s_{ik}(x_{ij}-v_{kj})^2$. This leads to the partition $\mathbf{w}^{(c+1)}$ defined by $w_{j\ell}^{(c+1)}$ equal to 1 if $\ell = \operatorname{argmin}_{\ell=1,\ldots,m} T_{j\ell}^{(c)}$ and 0 otherwise.

2) Computation of $\boldsymbol{\alpha}$ *given* $\mathbf{w}$*:* This step consists in minimizing H w.r. to $\boldsymbol{\alpha}$ given $\mathbf{w}^{(c+1)}$. Using the kth component and the ℓth cluster, the expression to minimize is

$$s_k^{(c)} w_\ell^{(c+1)} \log \sigma_{k\ell}^2 + \sum_{i,j} s_{ik}^{(c)} w_{j\ell}^{(c+1)} \frac{(x_{ij}-a_{k\ell})^2}{\sigma_{k\ell}^2}.$$

It leads to

$$a_{k\ell}^{(c+1)} = \frac{\sum_{ij} s_{ik}^{(c)} w_{j\ell}^{(c+1)} x_{ij}}{s_k^{(c)} w_\ell^{(c+1)}},$$

and

$$(\sigma_{k\ell}^2)^{(c+1)} = \frac{\sum_{i,j} s_{ik}^{(c)} w_{j\ell}^{(c+1)} (x_{ij}-a_{k\ell})^2}{s_k^{(c)} w_\ell^{(c+1)}}.$$

Using the terms v_{kj} and f_{kj} previously defined, the center and the variance of each block take respectively the following forms

$$\frac{\sum_j w_{j\ell}^{(c)} v_{kj}}{s_k^{(c)} w_\ell^{(c)}}$$

and

$$\frac{\sum_j w_{j\ell}^{(c+1)}\left(f_{kj} + s_k^{(c)}(v_{kj}-a_{k\ell})^2\right)}{s_k^{(c)} w_\ell^{(c+1)}}.$$

Note that, in the M-step, computational shortcuts are performed on a reduced matrix using sufficient statistics v_{kj} and f_{kj} and therefore it is suitable for large data sets.

C. Properties of Block EM

This GEM algorithm will be called in the following Block EM algorithm (BEM). Let us recall that GEM has the same convergence properties that EM and, like EM, is known to converge slowly in some situations. The second important drawback of these kind of algorithms is that their solutions can highly depend on its starting position and consequently produce sub-optimal maximum likelihood estimates. To act against this high dependency on its initial position, we propose to use the "em-EM" strategy which consists in several short runs of BEM from random positions followed by a long run of BEM from the solution maximizing the likelihood.

V. Block CEM algorithm

Regarding the context of clustering with the ML approach, after we estimate parameter $\boldsymbol{\theta}$, we can give a probabilistic clustering of the n objects in term of their fitted posterior probabilities of component membership s_{ik} obtained at the end of EM. Then, we can obtain a partition by using classification step which assigns each object to the component of the mixture to which it has the highest posterior of probability of belonging. With the optimal $\mathbf{w}$ partition, we obtain therefore a co-clustering where a partition of objects is characterized by a partition of variables. The BEM algorithm can be viewed as a soft algorithm to cluster simultaneously the set of objects and the set of variables.

A hard version called, Classification BEM, can be performed by replacing L($\boldsymbol{\theta}$) by $L_c(\mathbf{z}, \mathbf{w}; \boldsymbol{\theta})$. The main modifications concern the conditional maximization of complete data log-likelihoods w.r. to $\mathbf{w}$ given $\mathbf{z}$ and $\boldsymbol{\theta}$ and w.r. to $\boldsymbol{\theta}$ given $\mathbf{z}$ and $\mathbf{w}$. This leads to convert the posterior probabilities s_{ik}'s to a discrete classification ($z_{ik}^{(c)} = 1$ if $k = \text{argmax}_{k'=1,\ldots,g}\ s_{ik'}^{(c)}$ and $z_{ik}^{(c)} = 0$ otherwise) in a C-step before performing the M-step based this time on the clusters.

From these models we can impose that the proportions are equal and all blocks have the same variance. Then the complete data log-likelihood is equal to

$$-n \log g - \frac{nd}{2} \log \sigma^2 - \frac{1}{2\sigma^2} ||\mathbf{x} - \mathbf{zaw}^T||^2$$

then the maximization of L_c and the minimization of (2) are equivalent. We have a signification of criterion optimized by *Croeuc*: the proportions are supposed equal and the variances for all the blocks are the same. Moreover, *Croeuc* appears as a particular hard version of BEM.

VI. 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. Firstly, to demonstrate the advantage of BEM, we compared its performances with classical EM on the diagonal Gaussian model ignoring the clustering of variables. Secondly, we evaluate BEM when number of columns is higher that the rows. Thirdly, from data matrices non-negative, we study the performances of BEM versus the non-negative block decomposition (NBVD) in clustering context.

A. BEM versus Croeuc and EM

To illustrate the behavior of BEM, we selected a 1000×50 data arising from 3×2-component mixture model corresponding to three degrees of overlap of the clusters: well separated, moderately separated and poorly separated. The concept of cluster separation is difficult to visualize easily for our model, but the degree of overlap can be measured by the true error rate approximated by comparing the partitions simulated with those we obtained by applying a classification step. From our numerical experiments, we present only 3 situations corresponding to 3 levels of overlap degrees: M1 for clusters well separated (8.6%), M2 for moderately separated (16%) and M3 for poorly separated (24.8%). To compare two partitions $\mathbf{z}$ and $\mathbf{z}'$ having the same number of clusters, the error rate or the proportions of misclassified objects is noted $\delta(\mathbf{z}, \mathbf{z}')$. 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 I, we compared the performances of BEM, EM and *Croeuc* by using $\delta(\mathbf{z}, \mathbf{z}')$ (in percent) and their execution times recorded from the same initial positions. It appears clearly that BEM outperforms EM and *Croeuc*. In the other hands, BEM is more faster than EM, the rate timeEM/timeBEM noted $tEM/tBEM$ is higher than two. Different Monte Carlo simulations were performed confirming these remarks and also the superiority of BEM as compared to EM and *Croeuc*.

TABLE I
Comparison results between BEM, EM and *Croeuc*
($n \times d = 1000 \times 50$)

Error (%)	Situ.	BEM	EM	Croeuc	$\frac{tEM}{tBEM}$
$\delta(\mathbf{z}, \mathbf{z}')$	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

B. Effect of the size of data

Now we illustrate the interest of our approach when $n < d$, crucial situation in bioinformatics. As our model is parsimonious, it does not suffer of this situation and therefore offers a good alternative in order to cluster objects. The Table II, displays the degree of overlap and the error rates $\delta(\mathbf{z}, \mathbf{z}')$ for different sizes of n. We note incontestably that when $n < d$, BEM is always the best even if this superiority decreases naturally when $n > d$.

TABLE II
BEM vs EM when $n < d = 400$

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

C. BEM versus NBVD

In this paragraph, we simulated 1500×1000 non-negative data arising from 3×3-component mixture model with two degree of overlap (data1 and Data2). We have performed different experiences with NBVD et we remarked that, even it leads good approximations (not reported here), it has difficulties to give good partitions. As this weakness is due to the initialization of NBVD by arbitrary matrices $\mathbf{r}$, $\mathbf{a}$ and $\mathbf{c}$, we propose to initialize NBVD by the results of BEM. In other

words, $\mathbf{r}$, $\mathbf{a}$ and $\mathbf{c}$ are initialized by $\mathbf{z}$ (NBVD1) or $\mathbf{s}$ (NBVD2), $\mathbf{a}$ and $\mathbf{w}$ obtained by BEM.

The results of different data sets with different degree overlap are reported in respectively to the initialization of $\mathbf{r}$ by $\mathbf{z}$ or by $\mathbf{s}$. In Tables III, IV are reported confusion matrices (conf.rows for rows and conf.columns for columns) from the original data Data1 and Data2, and those obtained by NBVD. In clustering context, the initialization of NBVD by BEM appears more interesting for NBVD and in this case, it is slightly advantageous to initialize $\mathbf{r}$ by $\mathbf{z}$. However, we note that NBVD does not seem improving the obtained clustering by BEM. Hence, when the aim is co-clustering, BEM appears sufficient.

From BEM, we can define two criteria $||\mathbf{x} - \mathbf{zaw}^T||^2$ and $||\mathbf{x} - \mathbf{saw}^T||^2$. The criteria in NBVD are improved in both cases (see Tables V and VI); we have a better approximation when $\mathbf{r}$ is initialized by $\mathbf{s}$.

TABLE III
NBVD VS BEM FOR DATA1

Algo.	conf.rows	conf.columns
BEM	$\begin{pmatrix} 509 & 0 & 0 \\ 0 & 478 & 4 \\ 0 & 14 & 495 \end{pmatrix}$	$\begin{pmatrix} 317 & 0 & 0 \\ 0 & 343 & 0 \\ 0 & 0 & 340 \end{pmatrix}$
NBVD1	$\begin{pmatrix} 509 & 0 & 0 \\ 0 & 478 & 4 \\ 0 & 14 & 495 \end{pmatrix}$	$\begin{pmatrix} 317 & 0 & 0 \\ 0 & 343 & 0 \\ 0 & 0 & 340 \end{pmatrix}$
NBVD2	$\begin{pmatrix} 509 & 0 & 0 \\ 0 & 475 & 10 \\ 0 & 17 & 489 \end{pmatrix}$	$\begin{pmatrix} 317 & 0 & 0 \\ 0 & 343 & 0 \\ 0 & 0 & 340 \end{pmatrix}$

TABLE IV
NBVD VS BEM FOR DATA2

Algo.	conf.rows	conf.columns
BEM	$\begin{pmatrix} 492 & 1 & 16 \\ 1 & 460 & 0 \\ 22 & 0 & 488 \end{pmatrix}$	$\begin{pmatrix} 344 & 0 & 0 \\ 0 & 331 & 0 \\ 0 & 5 & 320 \end{pmatrix}$
NBVD1	$\begin{pmatrix} 492 & 1 & 16 \\ 1 & 460 & 0 \\ 22 & 0 & 488 \end{pmatrix}$	$\begin{pmatrix} 344 & 0 & 0 \\ 0 & 331 & 0 \\ 0 & 5 & 320 \end{pmatrix}$
NBVD2	$\begin{pmatrix} 492 & 0 & 17 \\ 3 & 461 & 1 \\ 20 & 0 & 486 \end{pmatrix}$	$\begin{pmatrix} 344 & 0 & 0 \\ 0 & 331 & 0 \\ 0 & 5 & 320 \end{pmatrix}$

TABLE V
NBVD VS BEM FOR DATA1

Algo.	$\|\|\mathbf{x} - \mathbf{zaw}^T\|\|^2$	$\|\|\mathbf{x} - \mathbf{saw}^T\|\|^2$
BEM	3.0015e+07	3.0010e+07
NBVD	2.9965e+07	2.9957e+07

VII. CONCLUSION

For co-clustering continuous data, we have proposed a new parsimonious mixture model. Contrary to latent block model requiring a variational approximation for binary data and co-occurrence data matrix, the proposed algorithm estimating the parameters of our model is a Generalized EM. It appears efficient and suitable for high-dimensional data.

TABLE VI
NBVD VS BEM FOR DATA2

Algo.	$\|\|\mathbf{x} - \mathbf{zaw}^T\|\|^2$	$\|\|\mathbf{x} - \mathbf{saw}^T\|\|^2$
BEM	4.5029e+07	4.5019e+07
NBVD	4.4957e+07	4.4936e+07

In hard clustering context: 1) we have given a sense of the criterion commonly used 2) we have proposed a general criterion taking into account the proportions of clusters and the variances of each block 3) we have shown that *Croeuc* or equivalent algorithms are hard versions of BEM and finally we have shown that even if the approximation of data non-negative is interesting with NBVD, its use without appropriated initializations cannot give good co-clustering.

Acknowledgments. This research was supported by the CLasSel ANR project ANR-08-EMER-002.

REFERENCES

[1] D. Baier, W. Gaul, M. Schader, *Two-mode overlapping clustering with applications to simultaneous benefit segmentation and market structuring*. In: Klar R, Opitz O (eds) Classification and knowledge organization. Springer, Heidelberg, 577–566, 1997

[2] Y. Chenga and G. Church, *Biclustering of expression data*, ISMB 2000, 8th International Conference on Intelligent Systems for Molecular Biology, 93–103, San Diego, California, 2000.

[3] H. Cho, I. Dhillon, Y. Guan and S. Sra, *Minimum Sum-Squared Residue Co-clustering of Gene Expression Data*, Proceedings of the Fourth SIAM International Conference on Data Mining, 114–125, 2004.

[4] G. Celeux and G. Govaert, *A Classification EM Algorithm for Clustering and two Stochastic Versions*, Computational Statistics and Data Analysis, 14, 315–332, 1992.

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

[6] I. Dhillon, *Co-clustering documents and words using bipartite spectral graph partitioning*, ACM SIGKDD International Conference, San Francisco, USA, 269–274, 2001.

[7] I. Dhillon, S. Mallela and D. S. Modha, *Information-Theoretic Co-clustering*, ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 89–98, 2003.

[8] G. Govaert, *Simultaneous Clustering of Rows and Columns*,Control and Cybernetics, 24,437-458,1995.

[9] G. Govaert and M. Nadif, *Clustering with block mixture models*, Pattern Recognition, 36, 463–473, 2003.

[10] G. Govaert, M. Nadif, *An EM algorithm for the block mixture model*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 27, 643–647, 2005.

[11] G. Govaert and M. Nadif, *Block clustering with Bernoulli mixture models: Comparison of different approaches*, Computational Statistics and Data Analysis, 52, 233–3245, 2008.

[12] G. Govaert and M. Nadif, *Latent Block Model for contingency table*, Communications in Statistics, Theory and Methods, 39, pp 416–425, 2010.

[13] G. J. McLachlan and D. Peel, *Finite Mixture Models*, Wiley, New York 2000.

[14] T. Li, and S. Ma, *IFD: Iterative Feature and Data clustering*,SIAM international conference on Data Mining (SDM) 536-543, 2004.

[15] T. Li, *A General Model for Clustering Binary Data*, KDD'05, 188–197, 2005.

[16] B. Long, Z. Zhang, Ph. S. Yu, *Co-clustering by value decomposition*, KDD'05,635–640,2005.

[17] W. Xu, X. Liu, and Y. Gong, *Document Clustering Based on Non-negative Matrix Factorization*, SIGIR'03, 267–273, 2003.