Position cluster latent block model for binary tables

Abstract— **Co-clustering methods are parsimonious approches for the analysis of block matrices by a simultaneous partitioning of the rows or the columns. Bringing the property of visualization to co-clustering is of first importance for a fast access to the essential topics and their relations for knowledge discovery for instance. A new generative method is proposed for the nonlinear mapping and the co-clustering of binary data by a particular parameterization of the Bernoulli latent block model. The probabilities in the Bernoulli distributions for the cells depend on bivariate coordinates which are the positions of the clusters. Experiments confirm the interest of our approch for several matrices from real data.**

Index Terms— **Latent Block Mixture Model, Binary Matrix, Block EM algorithm, Asymmetric Model , Bernoulli distribution**

I. Introduction

For data analysis and visualization of binary matrices, an extensive literature exists and diverse approaches have been developped until today. Some are based on a matricial decomposition or a distributional model but their linear setting can be a limiting factor. It is therefore interesting to look for an alternative approach with more flexibility in its underlying foundation. Today, the number of variables in a data matrix can be large because they code contents from textual, social or biological sources for instance with an increasing amount of data storage. Instead of focusing the modelling at the level of the datum [1], it is possible to deal with a partitioning in order to beneficiate from its parsimony. Some methods are based on a clustering procedure with vicinity constraints which are imposed on the mean centers. There exists in particular the family of methods called self-organizing maps which generalizes the Kohonen's map[2]. The later one is a variant of the k-means method which integrates topographic constraints into the centers in order to induce their self-organization in the data space. In a generative setting, a SOM has been formulated [3] with a Gaussian mixture model [4], [5], [6]. This method called Generative Topographic Mapping (GTM) includes neighborhood constraints at the level of the center means with the help of a common loading matrix and a common spherical covariance matrix for the noises. It is based on a well-defined criterion and the estimation of the parameters is performed by an EM algorithm [7] which guarantees the convergence.

For a matrix with a block structuce, a co-clustering is dramatically more efficient than a clustering of the two dimensions of the matrix separately because the clustering of one dimension is involved in the clustering of the other dimension, instead of handling them separately. When the data matrix is defined on a set I of objects (rows, observations) and a set J of variables (columns, attributes), co-clustering methods, in contrast to the usual clustering methods, consider the two sets I and J simultaneously [8], [9], [10]. Recently, a parsimonious model called block mixture model has been proposed [11], [12] by embedding co-clustering in a full generative setting. The developed algorithms are more efficient than a clustering applied separately on I and J [13].

The visualization of matrices can combine the block mixture model and constraints. The latent block model (LBM) is considered for the clustering part and the position clusters are introduced in the Bernoulli parameters by bringing parsimony. The new model is called Position Cluster Latent Block Model (PCLBM) and developped herein (see also [14]). It is an alternative to the topographic block latent model or block generative topographic mapping [15] which has been proposed for coupling self-organizing map with co-clustering in a full generative setting. The new approach is related to the latent position cluster model (LPCM) [16] which has been recently proposed for graph visualization with a multivariate Bernoulli mass distribution. Even if a variational expectation-maximization algorithm [17] exists for a faster inference of this model in comparison to the original implementation with a markov chain monte carlo, it is possible to look for an alternative approach which is dedicated to the clustering purpose. Our proposal can be seen as an asymmetric latent position cluster model with a clustering which is directly included in the model instead of the prior.

Note that the projection by re-parameterizing the latent block model is an alternative to a projection using the unconstrained latent block model via the construction of a distance matrix followed by a non parametric method such as a multidimensional scaling. Our parameterization is based on only one method in comparison to a chaining of several methods with a metric for the distance matrix which may be tricky to define in practice. With an ideal distance matrix and an ideal projection method, the result may be more relevant but this supposes several crucial steps which needs trials and validations in comparison to a direct parameterization of the model as proposed herein. This also leads to evaluate the quality of the nonlinear mapping for the comparison, this may be a matter of choices and purposes. Note that the parsimony of the clustering model can be mirrored with recent landscape methods in non parametric approaches: they use a limited set of data as representants of the whole dataset in order to reduce the numerical burden and treat directly n bidimensional positions for all the data vector insteads of g bidimensional positions for a small number of clusters but again this adds another step. For all these reasons, we prefer the presented approach even if this other way is an interesting perspective.

In this paper, section 2 presents our new method for visualization of binary data where the clustering foundation

is a Bernoulli block mixture model. To give the necessary background of the co-clustering approach under the mixture model, the block mixture model is also briefly reviewed. Section 3 focuses on the algorithms for the inference of the parameters involved in the mapping. Section 4 discusses the resulting mapping and the doule projection. Section 5 illustrates our method with several binary datasets by empirical experiments. Finally, the last section summarizes the main points of the contribution and ends with perspectives.

II. Parameterization of LBM with $2d$ coordinates

In this section, a new model is proposed by adding constraints to the Bernoulli latent block model. The inference of the parameters is a Generalized EM algorithm. Contrary to self organizing maps, the clusters are not arranged in a regular grid but are free to have their position anywhere on the latent space where the data are projected. The natural classes are also modeled with a smaller number of clusters in our experiments.

A. Latent block mixture model

In co-clustering, the $n \times d$ data matrix is defined by $\mathbf{x} = \{(x_{ij}); i \in I \text{ and } j \in J\}$ where $x_{ij} \in \mathbb{R}$. The aim of co-clustering is to summarize this matrix by homogeneous blocks. This problem can be studied under the simultaneous partition approach of two sets I and J into g and m clusters respectively. In [11], [12], the methods from [9], [10] have been modelled in the mixture approach. Algorithms for binary or fuzzy assignments to row clusters and column clusters have been proposed. In the context of co-clustering, the formulation of the mixture model can be extended [11] to propose a latent block model defined in particular by the following law by summing over all the assignments $\mathcal{Z} \times \mathcal{W}$:

$$f(\mathbf{x};\boldsymbol{\theta}) = \sum_{(\mathbf{z},\mathbf{w})} \prod_{i,k} p_k^{z_{ik}} \prod_{j,\ell} q_\ell^{w_{j\ell}} \prod_{i,j,k,\ell} \varphi(x_{ij};\alpha_{k\ell})^{z_{ik}w_{j\ell}} .$$

Here a set of assignments of $I \times J$ is defined by a product of assignments of I and J which are assumed to be independent. It is denoted $\mathcal{Z}$ and $\mathcal{W}$ for the sets of all possible assignments $\mathbf{z}$ of I and $\mathbf{w}$ of J. As in latent class analysis, the $n \times d$ random variables generating the observed cells x_{ij} are assumed to be independent once $\mathbf{z}$ and $\mathbf{w}$ are fixed. And, $\varphi(.;\alpha_{k\ell})$ is a law defined on the real set $\mathbb{R}$ while $\alpha_{k\ell}$ is an unknown parameter. The parameter $\boldsymbol{\theta}$ is compound of $\boldsymbol{\alpha} = (\alpha_{11},\ldots,\alpha_{gm})$, $\mathbf{p}$ and $\mathbf{q}$. Here, $\mathbf{p} = (p_1,\ldots,p_g)$ and $\mathbf{q} = (q_1,\ldots,q_m)$ are the vectors of probabilities p_k and q_ℓ that a row and a column belong to the k^{th} component and to the ℓ^{th} component respectively.

For binary data where $x_{ij} \in \{0,1\}$, it is defined a Bernoulli block mixture model [11] with parameters $\alpha_{k\ell} \in [0;1]$ and when it is denoted $\tau = \alpha_{k\ell}$, the following law for the cells:

$$\varphi(x_{ij};\tau) = (\tau)^{x_{ij}}(1-\tau)^{1-x_{ij}} .$$

Next, we propose a particular parameterization of the quantities $\alpha_{k\ell}$ and an algorithm for the parameters infererence by maximum likelihood.

B. Parameterization with latent coordinates

Herein, the parameters $\alpha_{k\ell}$'s of the block mixture model are parameterized with two sets of vectors. It is defined new coordinates in R^2, denoted $\boldsymbol{\xi}_k^r$ for the rows, and $\boldsymbol{\xi}_\ell^c$ for the columns.

Let's have β stands for a real scalar in R. Thus, it can be defined new parameters of the Bernoulli function with a sigmoidal function $\sigma(.)$ as follows :

$$\alpha_{k\ell} = \frac{exp(a_{k\ell})}{1+exp(a_{k\ell})} ,$$

where,

$$a_{k\ell} \quad = \quad \beta - |\boldsymbol{\xi}_k^r - \boldsymbol{\xi}_\ell^c|^2 = \beta - \sum_{s=1}^{2} (\boldsymbol{\xi}_{s,k}^r - \boldsymbol{\xi}_{s,\ell}^c)^2 .$$

By this way, each row and column cluster has its position on the plane. Note that alternative distance are possible for replacing the Euclidian one, such as the inner product or the absolute distance with the squared root.

Hence, the new set of parameters is is $\boldsymbol{\theta} = \{\beta, \boldsymbol{\Phi}_r, \boldsymbol{\Phi}_c\}$, where the $g \times m$ matrix $\boldsymbol{\alpha}$ is replaced by the matrices:

$$\begin{aligned} \boldsymbol{\Phi}_r &= (\boldsymbol{\xi}_{s,k}^r)_{2\times g} , \\ \boldsymbol{\Phi}_c &= (\boldsymbol{\xi}_{s,\ell}^c)_{2\times g} . \end{aligned}$$

Next, the objective function for the inference and a related algorithm are presented.

III. Objective function and inference

In this section, we are interested in the estimation of a value of $\boldsymbol{\theta}$ by maximizing the likelihood associated with the block mixture model. For this purpose we consider the Block Expectation-Maximization [12] or *block EM* (BEM) where the maximization step is adapted to the new central parameters.

A. Function $\tilde{Q}$

The completed data is the vector $(\mathbf{x},\mathbf{z},\mathbf{w})$ where the unobservable vectors $\mathbf{z}$ and $\mathbf{w}$ are the row and column labels. It is introduced the posterior probabilities c_{ik} that a data row i belongs to a row cluster k and $d_{j\ell}$ that a data column j belongs to a column cluster. Let's have $y_{k\ell}^{(t)} = \sum_{i,j} c_{ik}^{(t)} d_{j\ell}^{(t)} x_{ij}$, $u_{i\ell}^{(t)} = \sum_j d_{j\ell}^{(t)} x_{ij}$, $d_\ell^{(t)} = \sum_j d_{j\ell}^{(t)}$ and $v_{jk}^{(t)} = \sum_i c_{ik}^{(t)} x_{ij}$, $c_k^{(t)} = \sum_i c_{ik}^{(t)}$. Note also that the mixing probabilities are supposed equal. This results into the following criterion:

$$\begin{aligned} & \tilde{Q}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \\ = \; & \textstyle\sum_{k,\ell} y_{k\ell}^{(t)} x_{ij} a_{k\ell} - c_k^{(t)} d_\ell^{(t)} \log(1+e^{a_{k\ell}}) \\ + \; & \textstyle\sum_{i,k} c_{ik}^{(t)} \log(p_k) + \sum_{j,\ell} d_{j\ell}^{(t)} \log(q_l) . \end{aligned} \tag{1}$$

In the following, $p_k = 1/g$ and $q_\ell = 1/m$ without loss of generality. The optimization problem of this new objective function depending on $\{\boldsymbol{\xi}_k^r\}$ and $\{\boldsymbol{\xi}_\ell^c\}$ can be performed by the alternated maximization of conditional expectations like in BEM.

B. Expectation step

The posterior probabilities are found by solving the same problem than in the unconstrained case but with the new parameterization.

$$\begin{array}{rcl} c_{ik}^{(t)} & \propto & \prod_\ell (\sigma(a_{k\ell}^{(t)}))^{u_{i\ell}^{(t)}} (1-\sigma(a_{k\ell}^{(t)}))^{d_\ell^{(t)} - u_{i\ell}^{(t)}} , \\ d_{j\ell}^{(t)} & \propto & \prod_k (\sigma(a_{k\ell}^{(t)}))^{v_{jk}^{(t)}} (1-\sigma(a_{k\ell}^{(t)}))^{c_k^{(t)} - v_{jk}^{(t)}} . \end{array} \tag{2}$$

C. Derivatives for the inference

Let us denote $u_{k\ell} = y_{k\ell} - c_k d_\ell \alpha_{k\ell}$ and $v_{k\ell} = c_k d_\ell \alpha_{k\ell}(\alpha_{k\ell} - 1)$, and the first order and second order of the derivative of $\tilde{Q}$ with respect to β, as g_β and H_β respectively. They are written in closed-form:

$$g_\beta = \textstyle\sum_{k\ell} u_{k\ell} \text{ and } H_\beta = \sum_{k\ell} v_{k\ell} \, . \tag{3}$$

The derivatives with respect to the coordinates for the projection of the clusters have closed-form expressions. The gradient vectors $\mathbf{g}_k^r$ (resp. $\mathbf{g}_\ell^c$) and the Hessian matrices $\boldsymbol{H}_k^r$ (resp. $\boldsymbol{H}_\ell^c$) are for $\boldsymbol{\xi}_k^r$ (resp. $\boldsymbol{\xi}_\ell^c$). Note that $\boldsymbol{\xi}_{k\ell}^{rc}$ stands for $\boldsymbol{\xi}_k^r - \boldsymbol{\xi}_\ell^c$. The first order derivatives are:

$$\begin{array}{rcl} \mathbf{g}_k^r & = & -2 \sum_\ell u_{k\ell} \, \boldsymbol{\xi}_{k\ell}^{rc} \, , \\ \mathbf{g}_\ell^c & = & +2 \sum_k u_{k\ell} \, \boldsymbol{\xi}_{k\ell}^{rc} \, . \end{array} \tag{4}$$

The diagonal matrices in the Hessian the second order derivatives as follows:

$$\begin{array}{rcl} \boldsymbol{H}_k^r & = & -4 \sum_\ell v_{k\ell} \, \boldsymbol{\xi}_{k\ell}^{rcT} \boldsymbol{\xi}_{k\ell}^{rc} + \sum_\ell u_{k\ell} \mathbb{I}_2 \, , \\ \boldsymbol{H}_\ell^c & = & +4 \sum_k v_{k\ell} \, \boldsymbol{\xi}_{k\ell}^{rcT} \boldsymbol{\xi}_{k\ell}^{rc} - \sum_k u_{k\ell} \mathbb{I}_2 \, . \end{array} \tag{5}$$

Here, $\mathbb{I}_2$ is the bidimensional identy matrix. In practice, the Hessian are regularized by adding bivariate Gaussian priors to the log-likelihood with constant variances as explained next paragraph.

D. Training algorithm

For solving the nonlinear problem, the maximization step is a Newton-Raphson procedure which increases locally the objective function. When ϵ_{BEM} is a small constant equal to 10^{-7} for instance for the stopping criterion, the proposed algorithm is given in Figure (1). Note that the Hessian $\boldsymbol{H}_k^r$ and $\boldsymbol{H}_\ell^c$ matrices were regularized with the diagonal matrix $0.0001\mathbb{I}_2$ which comes from bivariate Gaussian priors for the latent positions, and H_β with the scalar 0.00001 while β is updated with the help of a proportional factor for correcting the eventual lack of approximation of the quadratic criterion. Moreover, each $\boldsymbol{\xi}_k^r$ and $\boldsymbol{\xi}_\ell^c$ were updated if and only if their particular update was able to increase the log-likelihood, otherwise they were kept equal to their value of the previous iteration. These numerical corrections introduced in the algorithm are mostly usual, this leads to a monotonous increase at each step of the function $\tilde{Q}$ in our experiments. At the end of the training stage when the relative variation of the function $\tilde{Q}$ is smaller than a constant ϵ_{BEM}, we end with parameters denoted with a hat, $\hat{\boldsymbol{\theta}}$.

- Initialization:
Initialize $\{c_{ik}^{(0)}\}$, $\{d_{j\ell}^{(0)}\}$, $\beta^{(0)}$, $(\boldsymbol{\Phi}_r^{(0)}, \boldsymbol{\Phi}_c^{(0)})$.

- E-Step:
Update $\{c_{ik}^{(t)}\}$ or $\{d_{j\ell}^{(t)}\}$ by (2), alternatively.

- M-Step:
Update $\boldsymbol{\theta}$ by considering the derivatives (3), (4), and (5) as:

$$\begin{array}{rcll} \beta & \leftarrow & \beta - \frac{1}{H_\beta} g_\beta \, , & \\ \boldsymbol{\xi}_k^r & \leftarrow & \boldsymbol{\xi}_k^r - (\boldsymbol{H}_k^r)^{-1} \mathbf{g}_k^r & \text{for } 1 \le k \le g \, , \\ \boldsymbol{\xi}_\ell^c & \leftarrow & \boldsymbol{\xi}_\ell^c - (\boldsymbol{H}_\ell^c)^{-1} \mathbf{g}_\ell^c & \text{for } 1 \le \ell \le m \, . \end{array}$$

- End:
If $\left| 1 - \frac{\tilde{Q}^{(t+1)}}{\tilde{Q}^{(t)}} \right| < \epsilon_{\text{BEM}}$ then stop else return to E-Step.

Fig. 1: Algorithm for learning the parameters in the model. figure

IV. Nonlinear mapping with $\hat{\boldsymbol{\theta}}$

For the construction of a map with the method, the sets of bivariate coordinates $\hat{\boldsymbol{\Phi}}_r$ leads to the projection of I and the sets of bivariate coordinates $\hat{\boldsymbol{\Phi}}_c$ leads to the projection of J.

A. Double projection

Which such constrained model, a nonlinear projection is generally defined by the average position of each row data i and each column variable with the contribution of the posterior probabilities, and also from equating with zero the final gradient vectors,

$$\begin{array}{rcl} \hat{\boldsymbol{\xi}}_{(i)}^r & = & \sum_k \hat{c}_{ik} \hat{\boldsymbol{\xi}}_k^r = \sum_\ell \left\{ \sum_k \frac{\hat{c}_{ik} \, \hat{u}_{k\ell}}{\sum_{\ell'} \hat{u}_{k\ell'}} \right\} \hat{\boldsymbol{\xi}}_\ell^c \, , \\ \hat{\boldsymbol{\xi}}_{(j)}^c & = & \sum_\ell \hat{d}_{j\ell} \hat{\boldsymbol{\xi}}_\ell^c = \sum_k \left\{ \sum_\ell \frac{\hat{d}_{j\ell} \, \hat{u}_{k\ell}}{\sum_{k'} \hat{u}_{k'\ell}} \right\} \hat{\boldsymbol{\xi}}_k^r \, . \end{array} \tag{6}$$

The expressions for the positions which are derived from the gradients make clearer the property of double projection. The position for the rows and the columns are written with only the positions for the row clusters or the position for the column clusters. But, the weights $\tilde{d}_{j\ell} = \sum_k \frac{\hat{c}_{ik} \, \hat{u}_{k\ell}}{\sum_{\ell'} \hat{u}_{k\ell'}}$, and $\tilde{c}_{ik} = \sum_\ell \frac{\hat{d}_{j\ell} \, \hat{u}_{k\ell}}{\sum_{k'} \hat{u}_{k'\ell}}$ may be difficult to interpret because they are not probabilities anymore and can be negative.

B. Post-treatment

The expressions in (6) raises a question on the relevance of such expression. Does the definition of the projection with the average values from the final center positions $\hat{\boldsymbol{\xi}}_k^r$ and $\hat{\boldsymbol{\xi}}_\ell^c$ leads to the best possible projection from the model.

The projection in (6) such usual approach might be not always relevant. For instance, with a separated class which is divided into two clusters, this projection leads to final positions belonging to a line segment. Similarly, there is no protection against non fuzzification which can lead also to unique final positions for a given whole cluster. These situations may be interesting in certain cases where one is mostly interested by a visualization of the clustering but this seems related to a simple

projection of the cluster centers and not a real projection of the data. In general it may be more relevant to have an access to the relative positions of the data via the projection. Hence, a suitable post treatement seems important here for a better as possible final map:

- A solution may be via the posterior probabilities c_{ik} which can be written with some bivariate positions $\tilde{\boldsymbol{\xi}}^r_{(i)}$ plus a quantity η related to a variance such as,

$$c_{ik} \propto exp\left(-\frac{\eta}{2}|\tilde{\boldsymbol{\xi}}^r_{(i)} - \boldsymbol{\xi}^r_k|^2\right) .$$

  The same would be for the columns actually. A justification of this expression is that in Gaussian mixture, (6) is just the update for the mean positions, as if the positions $\tilde{\boldsymbol{\xi}}^r_{(i)}$ were supposed fixed and already known. This leads to a possible additional constraint on the positions $\tilde{\boldsymbol{\xi}}^r_{(i)}$ via a penalization of the objective function.
- It can been added to the final position for the rows and the columns bivariate white noises with small variances:

$$\begin{array}{rcl} \hat{\boldsymbol{\xi}}^r_{(i)} & \leftarrow & \hat{\boldsymbol{\xi}}^r_{(i)} + \mathcal{N}^{\otimes 2}(0, 0.12) \\ \hat{\boldsymbol{\xi}}^c_{(j)} & \leftarrow & \hat{\boldsymbol{\xi}}^c_{(j)} + \mathcal{N}^{\otimes 2}(0, 0.12) . \end{array} \quad (7)$$

  Here, the additional quantities are independent bivariate centered normal noises with variances $0.12^2\,\mathbb{I}_2$.

This avoids unique positions in the case of binary classification matrices for instance. As a perspective, the posterior probabilities may be altered as explained for increasing how the points positions scatter around the centers in the visualization. The approach related to jittering was preferred in the experiments presented next section.

V. Numerical experiments

In this section, we experiment our new mapping method on several datasets in order to illustrate the approach. The approach is also compared to a binary gtm for validation of the results.

A. Datasets

The data are the same than in [15] excep the fifth one which gave very poor results here since the first experiments. The dataset $Bean4$ (or $B4$) is small soybean in *UCI Machine Learning Repository*. It is compound of 4 classes, for 47 soybeans and 35 original variables which lead to 37 binary variables after recoding categorical ones into dummy variables and removing constant ones. The dataset named $News4$ (or $N4$) in [18] is compound of 4 classes, for 400 textual documents and 100 terms. The dataset named $Classic3$ (or $C3$) in [19] is compound of 3 classes, for 3891 textual documents and 4303 terms. The dataset $Classic3 - small$ or *C3s* is a sample from $C3$ resulting to 450 documents and 171 words. Note that the textual matrices are binarized by truncating every co-occurrence greater to 1. The empirical output from the proposed algorithms are described next paragraphs.

B. Experimental settings

The experiments involved two steps after learning the parameters. The parameters and the estimated labels lead to a map and also several indicators. When a row $i \in I$ has a higher posterior probability $\hat{c}_{ik}$ for a cluster k then it belongs to this cluster and the label for the i^{th} row is estimated by $\hat{z}_i = k$ such as $\hat{z}_i = argmax_k\, \hat{c}_{ik}$. This leads also to a mapping where each datum is put inside the node with the label $\hat{z}_i$. Diverse indicators can then be computed.

The results of the methods are compared with two indicators. For an evaluation of the quality of the clustering, an error rate is obtained from the estimated labels $\{\hat{z}_i\}$ and denoted Error-rate. The error rate is the percentage of missclassified samples, say $\frac{\#\left\{z_i \neq \hat{z}^S_{\hat{z}_i}\right\}}{n}$. Here, $\hat{z}^S_k$ denotes the estimated class label by majority vote of the k^{th} cluster from the map while z_i is the true class label and $\#\{.\}$ is the cardinality of a set. This indicator may decrease with the size of the map g until a limit if the classes are not perfectly separated. For an evaluation of the quality of the nonlinear mapping, one indicator is obtained via a measure of the separation of the original classes from the true labels and the projection points $\{\hat{\boldsymbol{\xi}}_i\}$ in the latent space. The average of the Silhouettes [20], denoted S-index, is the mean value of $(b_i - a_i)/\max(a_i, b_i)$ where a_i and b_i are as follows. The first quantity is the average dissimilarity between the i^{th} datum and the other ones in the $\hat{z}_i^{\text{th}}$ class. The second one is the minimal average dissimilarity from the other classes. This indicator is confined in the interval $[-1; 1]$ and is preferred maximal for more compact classes.

A comparison is also proposed with the block generative topographic mapping where it has been chosen empirically the values for m and h after several trials according to the quality of the obtained map. For PCLBM, the number of column clusters have been kept the same than for the block generative topographic mapping, and the number of row clusters is chosen equal to twice the true number of classes which gave mostly the best results in practice.

C. Empirical results

The results from the proposed methods are summarized in the Table (I) and Table (II). They come from several runs of the propose algorithm called here PCLBM-EM for the inference of PCLBM. For each dataset, the best result is kept by selecting the ouput with the smaller error rate and highest S-index, these two events have generally occured at the same time. The results for the other approaches have been already commented in [15] with also a descriptive for the algorithms NR_1-EM, VR_1-EM, VR_2^d-EM, CASOM_B which are from variants of the generative topographic mapping for the three first ones and from a Bernoullian variant of the Kohonen's map for the fourth. In comparison the presented new parameterization is able to improve the visualization by increasing the S-index for all the datasets except $Bean4$ where the value is slighly the same. In the other hand, the clustering error increases by around one percent (and a half) for three datasets out of the four ones, and again $Bean4$ leads to similar results with a separation between the classes which is perfect. The mapping is really a double projection of the rows and columns

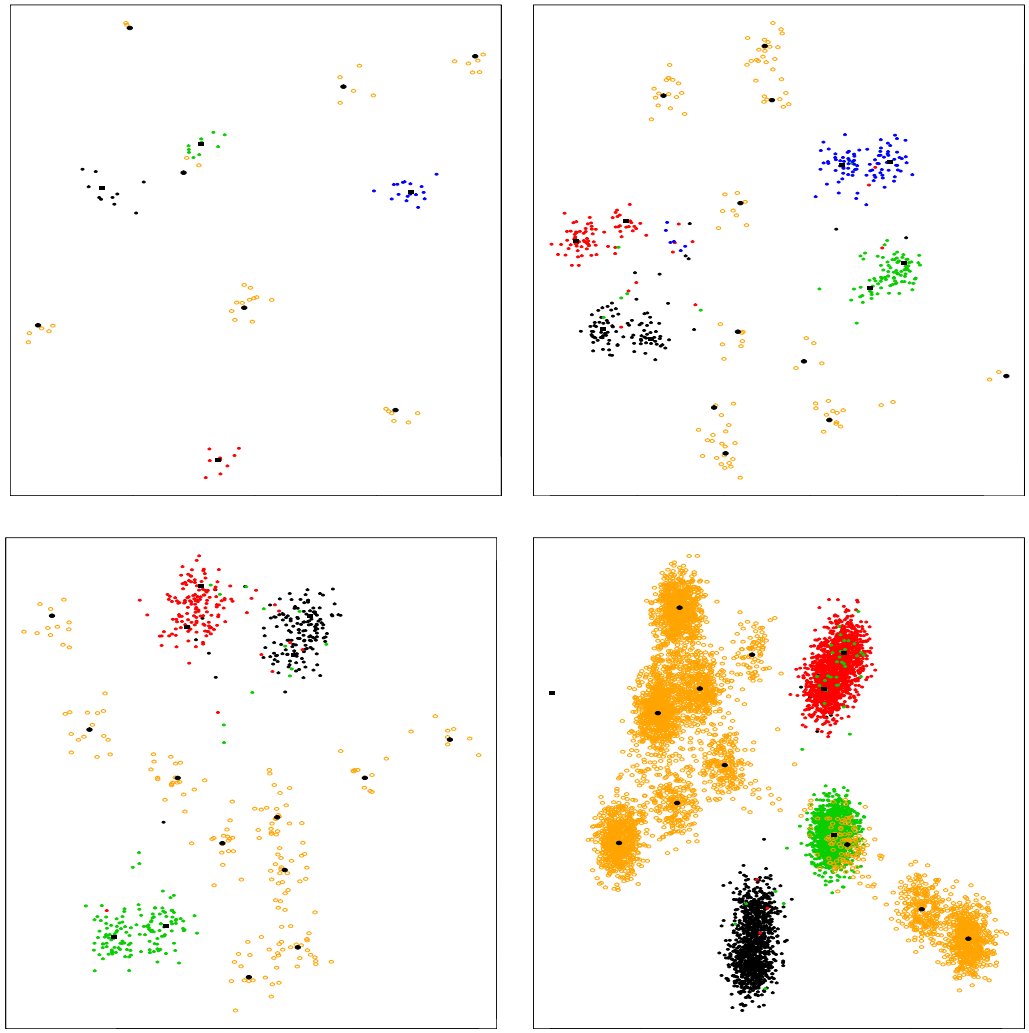

Fig. 2: Illustration of the nonlinear mapping PCLBM for the four datasets $B4$ (top-left), $N4$ (top-right), $C3s$ (bottom-left) and $C3$ (bottom-right).
figure

	B4	*N4*	*C3s*	*C3*
PCLBM-EM	0.00	06.72	05.78	01.67
NR_1-EM	0.00	05.12	04.45	01.11
VR_1-EM	0.00	04.58	04.68	01.36
VR_2^d-EM	0.00	06.47	04.23	–
$CASOM_B$	0.00	05.39	04.01	01.00

TABLE I: Error-rate in percent per method.
table

	B4	*N4*	*C3s*	*C3*
PCLBM-EM	0.84	0.56	0.62	0.73
NR_1-EM	0.82	0.51	0.48	0.64
VR_1-EM	0.85	0.52	0.48	0.64
VR_2^d-EM	0.50	0.35	0.34	–
$CASOM_B$	0.57	0.39	0.39	0.46

TABLE II: S-index per method.
table

of the binary variables. During the experiments it seems more interesting to have the points position of the columns in the middle of the projection, otherwise in certain cases, the two dimensions of the matrices are projected in separated areas of the plane and the error rate can increase, but this observation would need more investigation in future with other datasets for validation. It has been shown in a previous section that the point projections of the rows and the columns can be written as linear combinations of the same centers positions. But this is not in the same way, one is with positive bounded weights and the other one not, and they seem not be in the same projection space actually which is also observed visually with no real overlapping. Hence, the projections needs to be understand separately in the proposed model. Finally, it is clear that our proposal is able to lead to a fast overview of the true classes for these four datasets, with even less parameters than the block generative topographic mapping.

VI. Conclusion

Herein, we have proposed a model for the visualization of binary data by a new parameterization of the Bernoulli LBM. The approach can be also applied to graph visualization as a

particular case if the symmetry is modeled more explicitely. The new parameterization is able to handle a large number of variables when the data matrix can be partitioned with blocks. For helping revealing the true classes our approach appears as an appealing complement of the previous block generative topographic mapping and should be considered in practice as a complement for revealing visually the natural classes. Among future perspectives, this model could be extended to the case of coocurrence data in future with $e^{a_{k\ell}}$ for the parameterization. A bayesian inference is also of first interest in order to regularize automatically the parameters. The increase of the error rate may be reduced by introducing the information of nearest neighbors for instance in recent approaches. The case when the classes are less separated will require further experiments. Adding constraints for forcing the rows and the columns to project in a same space is also of main importance.

REFERENCES

[1] J. P. Benzecri, *Correspondence Analysis Handbook*, New-York : Dekker, 1992.

[2] Teuvo Kohonen, *Self-organizing maps*, Springer, 1997.

[3] Christopher M. Bishop, M. Svensén, and Christopher K. I. Williams, "Developments of the generative topographic mapping", *Neurocomputing*, vol. 21, pp. 203–224, 1998.

[4] M. J. Symons, "Clustering criteria and multivariate normal mixture", *Biometrics*, vol. 37, pp. 35–43, march 1981.

[5] G. J. McLachlan and K. E. Basford, *Mixture Models, Inference and applications to clustering*, Marcel Dekker, New York, 1988.

[6] Geoffery J. McLachlan and David Peel, *Finite Mixture Models*, John Wiley and Sons, New York, 2000.

[7] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum-likelihood from incomplete data via the EM algorithm", *J. Royal Statist. Soc. Ser. B., 39*, pp. 1–38, 1977.

[8] H. Bock, "Simultaneous clustering of objects and variables", in *Analyse des Données et Informatique*, E. Diday, Ed. 1979, pp. 187–203, INRIA.

[9] G. Govaert, *Classification croisée*, Thèse d'état, Université Paris 6, France, 1983.

[10] G. Govaert, "Simultaneous clustering of rows and columns", *Control and Cybernetics*, vol. 24, no. 4, pp. 437–458, 1995.

[11] Gérard Govaert and Mohamed Nadif, "Clustering with block mixture models", *Pattern Recognition*, vol. 36, no. 2, pp. 463–473, 2003.

[12] Gerard Govaert and Mohamed Nadif, "An EM algorithm for the block mixture model", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 4, pp. 643–647, 2005.

[13] Gerard Govaert and Mohamed Nadif, "Block clustering with bernoulli mixture models: Comparison of different approaches", *Computational Statistics & Data Analysis*, vol. 52, pp. 3233–3245, 2008.

[14] Rodolphe Priam and Mohamed Nadif, "Data visualization via latent variables and mixture models: a brief survey", *Pattern Analysis and Applications*, 2015.

[15] Rodolphe Priam, Mohamed Nadif, and Gérard Govaert, "Topographic bernoulli block mixture mapping for binary tables", *Pattern Analysis and Applications*, vol. 17, no. 4, pp. 839–847, 2014.

[16] Peter D. Hoff, Adrian E. Raftery, and Mark S. Handcock, "Latent space approaches to social network analysis", *Journal of the American Statistical Association, Theory and Methods*, vol. 97, no. 460, 2002.

[17] Michael Salter-Townshend and Thomas Brendan Murphy, "Variational bayesian inference for the latent position cluster model for network data", *Computational Statistics & Data Analysis*, vol. 57, no. 1, pp. 661–671, 2013.

[18] Ata Kabán and Mark Girolami, "A combined latent class and trait model for analysis and visualisation of discrete data", *IEEE Trans. Pattern Anal. and Mach. Intell.*, pp. 859–872, 2001.

[19] I. S. Dhillon, S. Mallela, and D. S. Modha, "Information-theoretic co-clustering", in *Proceedings of The Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD-2003)*, 2003, pp. 89–98.

[20] Peter Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis", *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, November 1987.