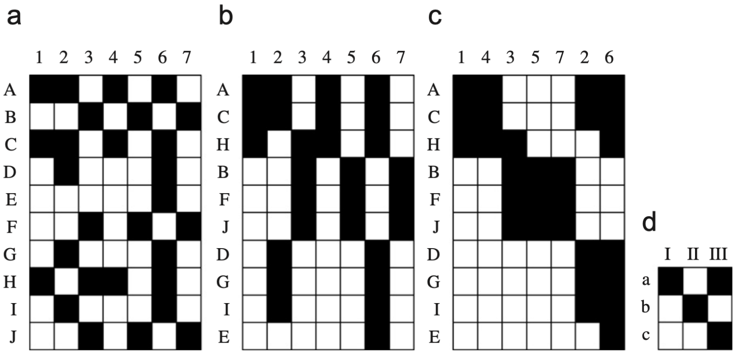
Model-based statistical learning: Co-clustering with the latent bloc model

Vincent Vandewalle (vincent.vandewalle@univ-cotedazur.fr)

Msc 2 Data-Science & IA, 2024-2025

UNIVERSITÉ CÔTE D'AZUR

Co-clustering aims at performing simultaneous clustering of both rows and columns:



Source: Christophe Biernacki, Julien Jacques, and Christine Keribin (2022). "A Survey on Model-Based Co-Clustering: High Dimension and Estimation Challenges". In

Bi-clustering, co-clustering and Latent Block Model (LBM)

- **Bi-clustering algorithms**: aim to detect homogeneous blocks within the data matrix which do not cover the entire matrix and which may overlap.
- **Co-clustering**: a specific bi-clustering model which assumes that all the individuals belong to one and only one row cluster, and *symmetrically* all the variables belong to only one column cluster.
- **Latent Block Model (LBM)**: LBM is a model for performing a model-based co-clustering

See Sara C Madeira and Arlindo L Oliveira (2004). "Biclustering algorithms for biological data analysis: a survey". In: *IEEE/ACM transactions on computational biology and bioinformatics* 1.1, pp. 24–45 for more details on bi-clustering algorithms.

Questions on Model-Based Clustering (MBC)

1. Recall the principle of model-based clustering
2. For what type of data is it designed?
3. What is the link between the component of the mixture and the clusters?
4. How to select the number of clusters?
5. How can your compare two partitions when performing clustering?
6. Why using the rand index?
7. Why performing only clustering on rows, then on columns would not be sufficient to solve the co-clustering problem?

Questions on Model-Based Clustering (MBC)

1. Recall the principle of model-based clustering Model the distribution of the data as a mixture of distributions.
2. For what type of data is it designed? Any kind of data as soon as we are able to propose a model for the class specific density.
3. What is the link between the component of the mixture and the clusters? Each component is interpreted as a cluster
4. How to select the number of clusters? It can be selected by AIC, BIC or ICL
5. How can your compare two partitions when performing clustering? By using the Adjusted Rand Index
6. Why using the rand index? It is invariant up to class permutation
7. Why performing only clustering on rows, then on columns would not be sufficient to solve the co-clustering problem? I allow to model the whole data matrix by a very sparse model.

The Latent Block Model (LBM) assumptions (1/2)

Data matrix $\mathbf{x}$ $(n \times d)$

- $\mathbf{x}_i$: the row/individual number i
- $\mathbf{x}^j$: the column/variable number j of $\mathbf{x}$
- x_i^j : variable j of individual i

Partition of the rows $\mathbf{z}$ $(n \times K)$

- $\mathbf{z} = (\mathbf{z}_1, \dots, \mathbf{z}_n)$
- $\mathbf{z}_i = (z_{i1}, \dots, z_{iK}) \in \{0, 1\}^K$
- $z_{ik} = 1$ if i belongs to row group k and 0 otherwise

Partition of the columns $\mathbf{w}$ $(d \times L)$

- $\mathbf{w} = (\mathbf{w}_1, \dots, \mathbf{w}_d)$
- $\mathbf{w}_i = (w_{j1}, \dots, w_{jL}) \in \{0, 1\}^L$
- $w_{j\ell} = 1$ if variable $\mathbf{x}^j$ belongs to column group ℓ and 0 otherwise

Main assumption: each point x_i^j is assumed to be independent given $\mathbf{z}_i$ and $\mathbf{w}_j$ (the knowledge of the block):

$$f(\mathbf{x}|\mathbf{z}, \mathbf{w}; \theta) = \prod_{k=1}^{K} \prod_{\ell=1}^{L} \prod_{i=1}^{n} \prod_{j=1}^{d} f(x_i^j; \alpha_{k\ell})^{z_{ik} w_{j\ell}}$$

with $f(\cdot; \alpha_{k\ell})$ the pdf associated to block $k\ell$ and parametrized by $\alpha_{k\ell}$.

The Latent Block Model (LBM) assumptions (2/2)

Moreover independence is assumed between all $\mathbf{z}_i$ and $\mathbf{w}_j$:

$$f(\mathbf{z}, \mathbf{w}; \theta) = \prod_{i,k} \pi_k^{z_{ik}} \prod_{j,\ell} \rho_j^{w_{j\ell}}$$

with $\pi = (\pi_k)_k$ (the probabilities of each cluster in row), $\rho = (\rho_\ell)_\ell$ (the probabilities of each cluster in column). $\theta = (\pi, \rho, \alpha)$ groups all the parameters

Thus

$$f(\mathbf{x}, \mathbf{z}, \mathbf{w}; \theta) = \prod_{i,k} \pi_k^{z_{ik}} \prod_{j,\ell} \rho_j^{w_{j\ell}} \prod_{i,j,k,\ell} f(x_i^j; \alpha_{k\ell})^{z_{ik} w_{j\ell}}$$

Marginalizing over $\mathbf{z}$ and $\mathbf{w}$ (since they are not observed in practice ...), the pdf of $\mathbf{x}$ is

$$f(\mathbf{x}; \theta) = \sum_{(\mathbf{z},\mathbf{w}) \in \mathcal{Z} \times \mathcal{W}} \prod_{i,k} \pi_k^{z_{ik}} \prod_{j,\ell} \rho_j^{w_{j\ell}} \prod_{i,j,k,\ell} f(x_i^j; \alpha_{k\ell})^{z_{ik} w_{j\ell}}$$

with $\mathcal{Z}$ (resp. $\mathcal{W}$) the set of all possible partitions of the rows (resp. the columns)

Choice of $f(\cdot;\alpha_{k\ell}$ according the type of data for x_i^j

- **Binary**: Bernoulli of parameter $\alpha_{k\ell}$
- **Categorical with r levels**: Multinomial distribution with parameters $\alpha_{k\ell} = (\alpha_{k\ell}^1, \ldots, \alpha_{k\ell}^r)$
- **Count data**: Poisson distribution with parameter $\alpha_{k\ell}$
- **Continuous**: Normal distribution with parameters $\alpha_{k\ell} = (\mu_{k\ell}, \sigma_{k\ell}^2)$
- Can be extended to numerous other data types (ordinal, functional, textual, ...)

These models are very parsimonious even in high dimension!
ToDo : Count the number of parameters of the LBM for each data type

LBM estimation

The observed log-likelihood is defined as:

$$\ell(\theta;\mathbf{x}) = \log f(\mathbf{x};\theta) = \log\left(\sum_{(\mathbf{z},\mathbf{w})\in\mathcal{Z}\times\mathcal{W}}\prod_{i,k}\pi_k^{z_{ik}}\prod_{j,\ell}\rho_j^{w_{j\ell}}\prod_{i,j,k,\ell} f(x_i^j;\alpha_{k\ell})^{z_{ik}w_{j\ell}}\right)$$

- $\ell(\theta;\mathbf{x})$ requires the computation of K^nL^d terms which correspond to all the possible configurations of unobserved labels $\mathbf{z}$ and $\mathbf{w}$!
- The problem is a missing data problem thus possible to use the EM algorithm

$Q(\theta;\theta')$ the expectation of the completed log-likelihood

- $\ell_c(\theta;\mathbf{x},\mathbf{z},\mathbf{w})$ the completed likelihood
- $Q(\theta,\theta') = \mathbb{E}(\ell_c(\theta;\mathbf{x},\mathbf{z},\mathbf{w});\mathbf{x},\theta')$ the expectation of the completed log-likelihood given the current parameters θ'

EM algorithm starting from $\theta^{(0)}$ and loop until convergence

- Expectation (E) step: Computation of $Q(\theta;\theta')$
- Maximization (M) step: $\theta^{(q+1)} = \arg\max_\theta Q(\theta,\theta^{(q)})$

E step: computation of $Q(\theta, \theta^{(q)})$

The EM algorithm allows to increase the log-likelihood at each iteration: $\ell(\theta^{(q+1)} \geq \ell(\theta^{(q)})$ and thus to converge to a local maximum of the likelihood

$$\ell_c(\theta; \mathbf{x}, \mathbf{z}, \mathbf{w}) = \sum_k (\sum_i z_{ik}) \log \pi_k + \sum_\ell (\sum_j w_{j\ell}) \log \rho_\ell + \sum_{i,j,k,\ell} \log f(x_i^j; \alpha_{k\ell})$$

Thus by taking the conditional expectation, we get:

$$\begin{aligned} Q(\theta, \theta^{(q)}) &= \sum_{i,k} p(z_{ik} = 1 | \mathbf{x}, \theta^{(q)}) \log \pi_k + \sum_{j,\ell} p(w_{j\ell} = 1 | \mathbf{x}, \theta^{(q)}) \log \rho_\ell \\ &+ \sum_{i,j,k,\ell} p(z_{ik} w_{j\ell} = 1 | \mathbf{x}; \theta^{(q)}) \log f(x_i^j; \alpha_{k\ell}) \end{aligned}$$

Let $s_{ik}^{(q)} = p(z_{ik} = 1 | \mathbf{x}; \theta^{(q)})$, $t_{j\ell}^{(q)} = p(w_{j\ell} = 1 | \mathbf{x}; \theta^{(q)})$ and $p(z_{ik} w_{j\ell} = 1 | \mathbf{x}; \theta^{(q)})$. All these computations are intractable due to dependence structure in the model.
Question: Assume that you would know these intractable quantities, how would perform the M-step?

Solution to the intractable E-step

- Variational approach: Constrain the joint probability to satisfy the relation

$$p(\mathbf{z}, \mathbf{w}|\mathbf{x}; \theta) \approx p_z(\mathbf{z}|\mathbf{x}; \theta) p_w(\mathbf{w}|\mathbf{x}; \theta)$$

  where p_z and p_w are chosen to provide the closest approximation of $p(\mathbf{z}, \mathbf{w}|\mathbf{x}; \theta)$ while still being computable. The algorithm maximizes an evidence lower bound (ELBO)

$$\ell(\theta; \mathbf{x}) \geq \mathcal{F}(\theta; \mathbf{x}) = \max_{p_z, p_w} (\ell_c(\theta; \mathbf{x}, \mathbf{z}, \mathbf{w}) - \log(p_z(\mathbf{z}) p_w(\mathbf{w}))$$

  this algorithm is called VEM as variational EM
- SEM algorithm : alternates the following steps: simulate $\mathbf{z}|\mathbf{x}, \mathbf{w}; \theta$ and then $\mathbf{w}|\mathbf{x}, \mathbf{z}; \theta$. Then update θ given the simulated classes $\mathbf{z}$ and $\mathbf{w}$

Estimating and evaluation of the rows and the columns clusters

Estimation

- VEM : based on $p_z(\mathbf{z}|\mathbf{x};\hat{\theta})$ and $p_w(\mathbf{w}|\mathbf{x};\hat{\theta})$ at the last iteration
- SEM: Based on sampling $(\mathbf{z},\mathbf{w})|\mathbf{x};\hat{\theta}$ by a Gibbs sampler, then estimate $(\hat{\mathbf{z}},\hat{\mathbf{w}})$ by the mode of the marginal sampled distribution.

Evaluation

- ARI: Adjusted Rand Rand Index / For the rows and columns respectively
- CARI: Co-clustering ARI developed for co-clustering