A Deep Dynamic Latent Block Model for the Co-clustering of Zero-Inflated Data Matrices

Giulia Marchello[✉][1], Marco Corneli[1,2], and Charles Bouveyron[1]

[1] Université Côte d'Azur, Inria, CNRS, Laboratoire J.A.Dieudonné, Maasai team, Nice, France.
[2] Université Côte d'Azur, Laboratoire CEPAM, Nice, France.

Abstract. The simultaneous clustering of observations and features of data sets (a.k.a. *co-clustering*) has recently emerged as a central machine learning task to summarize massive data sets. However, most existing models focus on stationary scenarios, where cluster assignments do not evolve in time. This work introduces a novel latent block model for the dynamic co-clustering of data matrices with high sparsity. The data are assumed to follow dynamic mixtures of block-dependent zero-inflated distributions. Moreover, the sparsity parameter as well as the cluster proportions are assumed to be driven by dynamic systems, whose parameters must be estimated. The inference of the model parameters relies on an original variational EM algorithm whose maximization step trains fully connected neural networks that approximate the dynamic systems. Due to the model ability to work with empty clusters, the selection of the number of clusters can be done in a (computationally) parsimonious way. Numerical experiments on simulated and real world data sets demonstrate the effectiveness of the proposed methodology in the context of count data.

Keywords: Co-clustering · Latent Block Model · zero-inflated distributions · dynamic systems · VEM algorithm.

1 Introduction

1.1 Context and related works

In a wide range of applications (e.g. signal processing, recommending systems, genetics, etc.) there is a growing need to develop machine learning models to treat time-dependent high dimensional data, in contexts of extreme data sparsity. By the simultaneous clustering of the the rows (observations) and the columns (features) of a data matrix, co-clustering proved to be an useful tool for high-dimensional data analysis thanks to its ability to provide useful summaries and visualisations of the data. However, the development of *dynamic* co-clustering methods for sparse data sets still remains almost an unexplored territory.

The cornerstone of model-based co-clustering is the popular latent block model (LBM, Govaert and Nadif, 2003), initially introduced for the co-clustering of binary data matrices. LBM is based on the assumption that rows and columns

of a matrix are grouped in hidden clusters and that the observations within a block (intersection of a row cluster and a column cluster) are independently and identically distributed. Whereas the original formulation of the model dealt with binary data only, the model has been extended in the last two decades to count data (Govaert and Nadif, 2010), continuous data (Lomet, 2012), categorical data (Keribin et al., 2015), ordinal data (Jacques and Biernacki, 2018; Corneli et al., 2020), functional data (Bouveyron et al., 2018) and textual data (Bergé et al., 2019). In the dynamic context, Boutalbi et al. (2020) proposed the tensor latent block model (TLBM) for the co-clustering of rows and columns of a 3D array, with covariates accounting for the third (temporal) dimension. TLBM was also implemented for different types of data: continuous, binary and counting. Recently, Marchello et al. (2022) proposed an extension of LBM allowing one to perform the simultaneous clustering of rows, columns and slices of a three dimensional counting array. Although being a first attempt to expand the LBM model to the dynamic case, this model has the limitation of not allowing cluster switches of rows/columns. In a different framework, Casa et al. (2021) prolong the latent block model to deal with longitudinal data, relying on the shape invariant model (Lindstrom, 1995). Boutalbi et al. (2021) developed a model-based co-clustering method for sparse three-way data, where the third dimension can be seen as a discrete temporal one. Here, the sparsity is handled following the same assumption as in Ailem et al. (2017) that all blocks outside the main diagonal share a common parameter.

1.2 Contribution of this work

The model that we introduce brings two major contributions in the field of dynamic co-clustering: first, observations (rows) and features (columns) are allowed to leave/join clusters over time; second, the data sparsity is explicitly taken into account by means of block dependent zero-inflated distributions. Before describing our model in more details, in the next section, we just point out the importance of the first contribution. Capturing the data dynamics is crucial in order to detect atypical phenomena that may have affected the underlying generative process. For instance, if at a given time t the value of some features suddenly increases for just one observation in a cluster, this suggests that the observation is likely to have switched to another cluster. A change point should be detected, leaving space for further analysis to inspect the causes. Thus, our aim was to develop a highly interpretable co-clustering method allowing practitioners to obtain faster visualizations of the results in order to automate the data analysis.

2 A Zero-Inflated dynamic LBM

The observed data are assumed to be collected into time evolving matrices, over the the interval $[0, T]$. We work in discrete time and assume that we have a time

partition of equally spaced points

$$0 = t_0 < t_1 < t_u \leq t_U = T.$$

Now up to rescaling, we can assume without loss of generality, that $t_{u+1} - t_u = 1$. Moreover, to simplify the exposition we omit the subscript u and, with a slight abuse of notation, we denote by t the generic time point t_u and by T the number of time points U. Thus, at (discretized) time t, we introduce the incidence matrix $X(t) \in \mathbb{N}^{N \times M}$ whose entry $X_{ij}(t)$ describes the (binary, counting, real) interaction between the observation i and the feature j took place between t and $t-1$. The rows of $X(t)$ are indexed by $i = 1, ..., N$ and the columns by $j = 1, ..., M$.

We aim at simultaneously clustering the rows and columns of the collection of the time indexed data matrices $\{X(t)\}_t$.

Cluster modeling. The rows (i.e. observations) and columns (i.e. features) of $X(t)$ are clustered into Q and L groups, respectively. Although Q and L are assumed fixed over time, each row/column is nevertheless allowed to change its cluster membership over $[0, T]$. More formally, a latent matrix $Z(t) := \{Z_{iq}(t)\}_{i \in 1,...,N; q \in 1,...,Q}$ represents the clustering of N *rows* into Q groups at a given time point t, with $Z_{iq}(t) = 1$ if row i belongs to the q-th cluster in t, zero otherwise. We assume that the i-th row of $Z(t)$ (say $Z_i(t)$) follows an evolving multinomial distribution, parameterized by $\alpha(t)$

$$Z(t) \sim \mathcal{M}(1, \alpha(t) := (\alpha_1(t), \ldots, \alpha_Q(t))), \tag{1}$$

where $\alpha_q(t) = \mathbb{P}\{Z_{iq}(t) = 1\}$ and $\sum_{q=1}^{Q} \alpha_q(t) = 1$, for all t.

In a similar fashion, we introduce a latent matrix $W(t) \in \{0, 1\}^{M \times L}$, labelling the column clusters at time t, and whose j-th row $W_j(t)$ follows a multinomial distribution of parameter $\beta(t) := (\beta_1(t), \ldots, \beta_L(t))$.

The two random matrices Z and W are further assumed to be independent.

Sparsity modeling. In order to model a potentially extreme data sparsity, the observed data are modeled by mixtures of block-conditional Zero-Inflated distributions, with conditionally independent entries $X_{ij}(t)$. In more detail we introduce a latent vector π of length T, whose entry $\pi(t)$ indicates the proportion of data sparsity at time t. Then we assume that, with probability $\pi(t)$, $X_{ij}(t) = 0$ a.s., whereas with probability $1 - \pi(t)$ we have[1]

$$X_{ij}(t) | Z_i(t), W_j(t) \sim \varphi(X_{ij}(t); \zeta_{Z_i(t), W_j(t)}), \tag{2}$$

independently for all (i, j), where $\varphi(X_{ij}(t), \cdot)$ is some probability distribution function with parameter $\zeta \in \mathbb{R}^{Q \times L}$. In a compact notation:

$$X_{ij}(t) | Z_i(t), W_j(t) \sim ZI_\varphi(\zeta_{Z_i(t), W_j(t)}, \pi(t)) \ , \tag{3}$$

[1] We adopt in Eq (2) a quite common convention in the clustering literature: $Z_i(t)$ denotes both the i-th row of $Z(t)$ and a random variable whose value is q if row i is in the q-th row cluster at time t.

where ZI stands for Zero-Inflated. Among the distributions $\varphi(\cdot)$ that could be considered, we can cite the zero-inflated versions of the log-normal and the Gamma distributions for continuous data, or the zero-inflated Poisson (ZIP) distribution (Lambert, 1992) for count data.

In order to ease the illustration of the inference routine we finally provide a third, equivalent formulation of the above equations in terms of a hidden random matrix, $A \in \{0,1\}^{N\times M}$, where independently for all i and j

$$A_{ij}(t) \sim \mathcal{B}(\pi(t)),$$

with $\mathcal{B}(p)$ denoting the Bernoulli probability mass function of parameter p and such that

$$\begin{aligned} &A_{ij}(t) = 1 \Rightarrow X_{ij}(t)|Z_i(t), W_j(t) = 0 \\ &A_{ij}(t) = 0 \Rightarrow X_{ij}(t)|Z_i(t), W_j(t) \sim \varphi(X_{ij}, (t), \zeta_{Z_i(t),W_j(t)}). \end{aligned} \tag{4}$$

Modeling the parameters dynamics. The mixing parameters α and β as well as the sparsity proportions π (all vectors of length T) are assumed to be driven by systems of ordinary differential equations (ODEs). In this way, we are able to capture the temporal evolution of both the cluster proportions and the (excess of) sparsity. In continuous time, the three dynamic systems would read as:

$$\frac{d}{dt}a(t) = f_Z(a(t)), \tag{5}$$

$$\frac{d}{dt}b(t) = f_W(b(t)), \tag{6}$$

$$\frac{d}{dt}c(t) = f_A(c(t)), \tag{7}$$

where $t \in [0,T]$, $f_Z : \mathbb{R}^Q \to \mathbb{R}^Q$, $f_W : \mathbb{R}^L \to \mathbb{R}^L$ and $f_A : \mathbb{R} \to \mathbb{R}$ are three unknown continuous functions and $a : [0,T] \to \mathbb{R}^Q$, $b : [0,T] \to \mathbb{R}^L$ and $c : [0,T] \to \mathbb{R}$ are three continuously differentiable functions such that

$$\alpha_q(t) := \frac{e^{a_q(t)}}{\sum_{q=1}^{Q} e_q^a(t)} \quad \beta_\ell(t) := \frac{e^{b_\ell(t)}}{\sum_{\ell=1}^{L} e_\ell^b(t)}, \tag{8}$$

and

$$\pi(t) := \frac{e^{c(t)}}{1+e^{c(t)}}. \tag{9}$$

Then, since (as stated at beginning of Section 2) we work with discrete time points, the above dynamic systems reduce to their Euler schemes. A graphical representation of the model described so far, and named Zero-Inflated dLBM, can be seen in Figure 1.

2.1 The joint distribution

The model described so far can be adapted to any zero-inflated distribution. The first formulation as well as the most well-known concerns the Zero-Inflated Poisson (Lambert, 1992). However, other distributions such as Zero-Inflated Negative

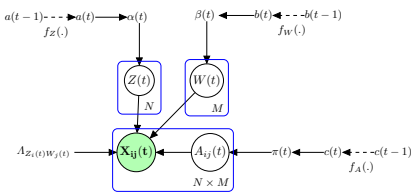

Fig. 1: Graphical representation of the Zero-Inflated dLBM model.

Binomial (Ridout et al., 2001), Zero-Inflated Beta (Ospina and Ferrari, 2012), Zero-Inflated log-normal (Li et al., 2011) could be coupled with the present modeling. In the following to ease the readability of the inference procedure we make use of the Zero-Inflated Poisson ($\text{ZI}_{\mathcal{P}}$) formulation to illustrate our approach. Hence, we can write

$$X_{ij}(t)|Z_i(t), W_j(t) \sim ZI_{\mathcal{P}}(\Lambda_{Z_i(t),W_j(t)}, \pi(t)),$$

where $\mathcal{P}(\cdot)$ denotes the probability mass function of a Poisson distribution and Λ is a $Q \times L$ matrix, denoting the block-dependent Poisson intensity parameter. The whole set of the model parameters is denoted by $\theta := (\Lambda, \alpha, \beta, \pi)$ and the latent variables used so far are A, Z and W. Thus, the likelihood of the complete data reads

$$p(X, A, Z, W|\theta) = p(X|A, Z, W, \Lambda, \pi) \times p(A \mid \pi)p(Z|\alpha)p(W|\beta). \tag{10}$$

The terms on the right hand side of the above equation can be further developed. Details are postponed in Appendix 1.1 for lack of space.

3 Inference

In order to infer the model parameters, two main problems occur. First, we can't adopt the EM algorithm (Dempster et al., 1977; Bishop, 2006) in order to numerically compute ML estimates from the intractable quantity $p(X|\theta)$. This issue is common to all stochastic and latent block models (see for instance Govaert and Nadif, 2003) due to the intractability of the posterior distribution of the latent variables (here A, Z and W). Second, although variational strategies (Jaakkola and Jordan, 1997; Jordan et al., 1998) could be employed, α, β and π cannot be updated explicitly, in the M step, due to the dynamics in Eqs. (5)-(7). This is why we combine variational inference with a Gradient Descent (GD) optimization for the ODE part.

3.1 Variational decomposition

Since we cannot compute the joint posterior distribution $p(A, Z, W|X, \theta)$, we introduce a variational distribution $q(\cdot)$ over the latent variables (A, Z, W) and

adopt the following standard variational decomposition of the observed log-likelihood

$$\log p(X|\theta) = \mathcal{L}(q;\theta) + KL(q(\cdot)||p(\cdot|X,\theta)),$$

where $\mathcal{L}$ denotes a lower bound of the term on the left hand side of the equality and is defined as:

$$\mathbb{E}_{q(A,Z,W)}\Big[\log \frac{p(X,A,Z,W|\theta)}{q(A,Z,W)}\Big] \tag{11}$$

and KL indicates the Kullaback-Liebler divergence between the approximate and the true posterior distribution of (A,Z,W). Although the above equations hold for any distribution $q(\cdot)$, we look for one that maximizes $\mathcal{L}(\cdot;\theta)$ (or equivalently, that minimizes the KL divergence) while keeping the maximization problem tractable. Hence, we adopt the following mean-field assumption

$$q(A,Z,W) = q(A)q(Z)q(W) = \prod_{i,j,t} q(A_{ij}(t)) \prod_{i,t} q(Z_i(t)) \prod_{j,t} q(W_j(t)). \tag{12}$$

Thus, we introduce a variational expectation-maximization algorithm that alternates an expectation step (VE) maximizing the lower bound in Eq. (11) with respect to the variational distribution $q(\cdot)$, while keeping θ fixed and a maximization step (VM), maximizing the lower bound $\mathcal{L}(q,\theta)$ with respect to $\theta = (\Lambda, \alpha, \beta, \pi)$, while holding the variational distribution $q(\cdot)$ fixed. The two steps are now described in much detail.

3.2 VE-Step

The optimal variational updates of $q(\cdot)$, under the assumption in Eq. (12), can be obtained as Bishop (2006):

$$\log q(A) := \mathbb{E}_{W,Z}[\log p(X,A,Z,W \mid \theta)], \tag{13}$$

$$\log q(Z) := \mathbb{E}_{A,W}[\log p(X,A,Z,W \mid \theta)], \tag{14}$$

$$\log q(W) := \mathbb{E}_{A,Z}[\log p(X,A,Z,W \mid \theta)]. \tag{15}$$

Optimization of q(A) The expectation in Eq. (13) can be explicitly computed leading to the following

Proposition 1. *Denoting by $\delta_{ij}(t) := q(A_{ij}(t) = 1)$ the variational probability of success for $A_{ij}(t)$, the optimal update is:*

$$\delta_{ij}(t) = \frac{\exp(R_{ij}(t))}{1 + \exp(R_{ij}(t))}, \tag{16}$$

with:

$$\begin{aligned} R_{ij}(t) := \log(\pi(t)\mathbf{1}_{\{X_{ij}(t)=0\}}) + \sum_{q,\ell} \Big[\mathbb{E}[Z_{iq}(t)]\mathbb{E}[W_{j\ell}(t)](\Lambda_{q\ell} + \\ - X_{ij}(t)\log\Lambda_{q\ell})\Big] + \log X_{ij}(t)! - \log(1-\pi(t)) \end{aligned} \tag{17}$$

where $\mathbf{1}_{\{\cdot\}}$ denotes the indicator function.

The proof is provided in the Appendix 1.2. Note that, formally, when $X_{ij}(t) \neq 0$, $R_{ij}(t) = -\infty$ and $\delta_{ij}(t) = 0$, which makes sense: non-null observations in X come from a Poisson distribution with probability one (see Eq. (4)).

Optimization of q(Z) and q(W) Regarding the factor $q(Z)$, the expectation in Eq. (14) can be explicitly computed leading to the following

Proposition 2. *Denoting by* $\tau_{iq}(t) := q(Z_{iq}(t) = 1)$ *the variational probability of success of* $Z_{iq}(t)$*, the optimal update is:*

$$\tau_{iq}(t) = \frac{r_{iq}(t)}{\sum_{v=1}^{Q} r_{iv}(t)}, \tag{18}$$

with

$$r_{iq}(t) \propto \exp\left(\sum_{j,\ell} F_{jl}^{iq} + \log(\alpha_q(t))\right) \tag{19}$$

and

$$F_{jl}^{iq} := (1 - \mathbb{E}[A_{ij}(t)])\Big[E[W_{j\ell}(t)](X_{ij}(t)\log(\Lambda_{q\ell}) - \Lambda_{q\ell})\Big]. \tag{20}$$

The proof is provided in the Appendix 1.3. In a similar way, for the factor $q(W)$, the expectation in Eq. (15) can be explicitly computed leading to the following

Proposition 3. *Denoting by* $\eta_{j\ell}(t) := q(W_{j\ell}(t) = 1)$ *the variational probability of success of* $W_{j\ell}(t)$*, the optimal update is:*

$$\eta_{j\ell}(t) = \frac{s_{j\ell}(t)}{\sum_{v=1}^{L} s_{jv}(t)}, \tag{21}$$

with

$$s_{j\ell}(t) \propto \exp\left(\sum_{i,q} G_{iq}^{j\ell} + \log(\beta_\ell(t))\right). \tag{22}$$

$$G_{iq}^{j\ell} := (1 - \mathbb{E}[A_{ij}(t)])\Big[E[Z_{iq}(t)](X_{ij}(t)\log(\Lambda_{q\ell}) - \Lambda_{q\ell})\Big]. \tag{23}$$

The proof is provided in the Appendix 1.4.

3.3 Variational M-Step

The lower bound can be explicitly computed as stated in Proposition 2 in Appendix 1.5 for lack of space. From that bound, we can optimize the model parameters θ, while keeping $q(\cdot)$ fixed, as stated in the reminder of this section.

Update of Λ We now report the update of the Zero-inflated Poisson parameter Λ. Note that in case other zero-inflated distributions are chosen, this step must be adapted to the corresponding distributions.

Proposition 4. *The updating formula of Λ is:*

$$\Lambda_{q\ell} = \frac{\sum_{i,j,t} \tau_{iq}(t)\eta_{j\ell}(t)\Big(X_{ij}(t) - \delta_{ij}(t)X_{ij}(t)\Big)}{\sum_{i,j,t} \tau_{iq}(t)\eta_{j\ell}(t)\Big(1 - \delta_{ij}(t)\Big)}. \tag{24}$$

The proof is provided in the Appendix 1.6. We just wish to point out that the above update formula is indeed very intuitive: it corresponds to a sample mean accounting for both the probability that null $X_{ij}(t)$s come from a Poisson distribution (via $1 - \delta_{ij}(t)$) and the probability that non-null $X_{ij}(t)$s come from co-cluster (q, l).

Update of α, β and π through deep neural networks The mixture proportions α and β, as well as the sparsity parameter π are driven by three systems of differential equations, in Eqs. (5),(6) and (7), respectively. As we assumed that the functions f_A, f_W and f_Z are continuous, we propose to parametrize them with three fully connected **neural networks** (Gent and Sheppard, 1992), with two hidden layers of 200 neurons each, equipped with ReLu activation functions, with parameters ω_A, ω_Z and ω_W, respectively. Thus, optimizing the lower bound $\mathcal{L}(q;\theta)$ with respect to α, β and π reduces to maximize it with respect to the parameters of the neural nets as well as to the initial values $a(0), b(0)$ and $c(0)$.

For $k \in \{A, Z, W\}$, if we denote by $\omega_k(h)$ the set of weights of the corresponding neural network at iteration h of the GD algorithm, then

$$\omega_k(h) = \omega_k(h-1) - \gamma \nabla_{\omega_k} \mathcal{L}, \tag{25}$$

where γ is a user defined learning rate, $\nabla_{\omega_k}(\cdot)$ is the gradient operator, with respect to ω_k and $\omega_k(0)$ is randomly sampled. In the experiments, this update is implemented in PyTorch via automatic differentiation (Paszke et al., 2017) and relies on stochastic optimisation (ADAM, Kingma and Ba, 2014). The learning rates are fixed at $\gamma = 1e^{-4}$. Once the neural nets are trained via back-propagation they provide us with the ML estimates of α, β and π. The inference procedure is summarized in Algorithm 1.

3.4 Initialization and model selection

When dealing with clustering methods based on the EM algorithm, the initialization and the selection of the appropriate numbers of clusters (for rows and columns here) are two issues which deserve an appropriate treatment. The issues related to these two points are slightly complicated here by the use of deep neural networks for modeling the dynamics of cluster and sparsity proportions. Despite this apparent difficulty due to the intrinsic complexity of these networks, they will nevertheless offer some unexpected flexibilities that we may use to lower

Algorithm 1 VEM-GD Algorithm (Zero-Inflated Poisson)

Require: $X, Q, L, n\_iter, nb\_epochs$ and $\alpha, \beta, \pi, \Lambda$ from Algorithm 2.
▶ Initialization of τ and η, sampled from $\mathcal{M}(N, \alpha)$ and $\mathcal{M}(M, \beta)$, respectively;
▶ Initialization of δ as ones(N, M), then setting $\delta_{ij} = 0$ when $X_{ij} > 0$;
while not $\mathcal{L}$ converges **do**
  VE-Step:
  for counter = 1 to n_iter **do**
    alternatively update δ, τ, η % *fix point eqs*
  end for
  M-Step:
  Update Λ via Eq. (24)
  Update α, β, π via ADAM % *over* nb_epochs
end while

the computational cost of the whole algorithm. Indeed, and as it is illustrated in the numerical experiments (Appendix 2), the use of deep neural networks for modeling the row and column cluster proportions will allow our algorithm to work with some empty clusters. Therefore, in the objective of avoiding the usual computationally demanding procedure of testing all pairs of row and column cluster numbers, we propose the following strategy for both initialization and model selection. First, we select a single specific slice of the data $X_{t_{init}}$ fit to it a static version of our $ZI_{\mathcal{P}}$-dLBM (technically a $ZI_{\mathcal{P}}$-LBM) for a list of pairs of cluster numbers, i.e. (q, ℓ) for $q = 2, \ldots, Q_{max}$ and $\ell = 2, \ldots, L_{max}$. We then use the ICL criterion (Integrated Classification Likelihood, Biernacki et al., 2000) to select the most appropriate row and column clusters' numbers for this specific slice of data. Let us remind that the ICL criterion aims at approximating the complete-data integrated log-likelihood and can be derived for $ZI_{\mathcal{P}}$-LBM as follows:

$$ICL(Q, L) = \log p(X, \hat{Z}, \hat{W}; \hat{\theta}) - \frac{Q-1}{2} \log N - \frac{L-1}{2} \log M - \frac{QL-1}{2} \log(NM). \quad (26)$$

The pair $(\hat{Q}, \hat{L})$ that leads to the highest value of the ICL is retained for the data $X_{t_{init}}$. Remark that, unless a further specific notice, the slice $X_{t_{init}}$ considered for this step in our experiments will be the first slice of the data, i.e. X_{t_0}. Second, in order to initialize our VEM-GD algorithm (see Algorithm 1) with useful initial values of the model parameters, we adopt a cascade process in order to propagate the parameters estimates obtained on the slice $X_{t_{init}}$ to other slices. In more detail, fixing for the moment the numbers of row and column clusters to $(\hat{Q}, \hat{L})$, we fit the static $ZI_{\mathcal{P}}$-LBM to the next slice $X_{t_{init}+1}$ with parameters $\hat{\theta}_{t_{init}}$ as initial values. Then, the estimated parameters $\hat{\theta}_{t_{init}+1}$ are used as initialization for a static $ZI_{\mathcal{P}}$-LBM fitted to the slice $X_{t_{init}+2}$, and so on up to X_T. This strategy allows us to obtain initial values (say $\hat{\theta}(t)$) for all the model parameters for $t = 0, ..., T$. Finally, as we expect that the choice of $\hat{Q}$ row and $\hat{L}$ column cluster components could not be the best when considering the data set as a whole, the VEM-GD algorithm (see Algorithm 1) is run with more components than considered in the initialization. Indeed, we run the VEM-GD algorithm

with $Q_{max} \geq \hat{Q}$ and $L_{max} \geq \hat{L}$ cluster components. Then, part of the model parameters are initialized with $\hat{\theta}(t)$ obtained via the initialization procedure described above (see Algorithm 2) and the remaining parameters, corresponding to the additional row and column clusters are set to zero. Thus, we aim at exploiting the potential "blessing" of the use of deep neural networks allowing our VEM-GD algorithm to start with some empty clusters. These empty clusters will have the possibility to be activated later in the inference process, if needed. Therefore, we avoid the usual computationally demanding procedure of running the whole algorithm with all pairs of row and column cluster numbers for the whole data set. This strategy allows our approach to scale to massive data sets in a reasonable computational time and with satisfying results, as shown in the next section.

Algorithm 2 Initialization

Step 1: Static model selection
Require: $X, Q_{min}, Q_{max}, L_{min}, L_{max}, max\_iter$, $n.sim.$
for Q =Q_{min}, to Q=Q_{max} **do**
for L =L_{min}, to L=L_{max} **do**
Initialize randomly $\alpha, \beta, \pi, \Lambda$;
Run $ZI_{\mathcal{P}}$-LBM on X_1 and compute ICL;
end for
end for
Ensure: (Q^*, L^*) that gives the highest ICL value.
Step 2: Cascade process
Require: X , Q^*, L^*, max_iter.
for t = 1 to T **do**
if t =1 **then**
Initialize randomly $\alpha, \beta, \pi, \Lambda$;
Run $ZI_{\mathcal{P}}$-LBM(Q^*, L^*) on X_1;
Store $\hat{\alpha}(1), \hat{\beta}(1), \hat{\pi}(1), \hat{\Lambda}$.
else
Initialize with $\hat{\alpha}(t-1), \hat{\beta}(t-1), \hat{\pi}(t-1), \hat{\Lambda}$;
Run $ZI_{\mathcal{P}}$-LBM(Q^*, L^*) on X_t;
Store $\hat{\alpha}(t), \hat{\beta}(t), \hat{\pi}(t), \hat{\Lambda}$.
end if
end for

4 Analysis of the adverse drug reaction dataset

In Appendix 2, there are in-depth experiments to verify the performances of the model on simulated data in different scenarios. This section focuses on the application of $ZI_{\mathcal{P}}$-dLBM to a large-scale pharmacovigilance data set, with the aim of illustrating the potential of the tool.

4.1 Protocol and data

This section considers a dataset consisting of an adverse drug reaction (ADR) data set, collected by the Regional Center of Pharmacovigilance (RCPV), located in the University Hospital of Nice (France). A time horizon of 7 years is considered, from January 1^{st}, 2015 to March 3^{th}, 2022, the unity measure for the time interval is a trimester. The overall dataset is made of 27,754 declarations, for which the market name of the drug, the notified ADR and the reception date are considered. Moreover, we only considered drugs and ADRs that were

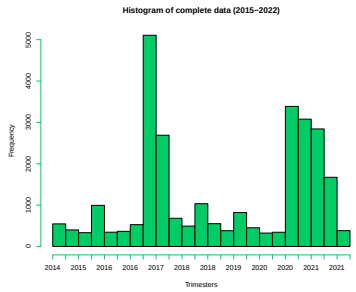

Fig. 2: Number of declarations received by the pharmacovigilance center from 2015 to 2022, sorted by trimester.

notified more than 20 times over the 7 years. The resulting dataset contains 236 drugs, 324 ADRs and 29 time intervals with 12,336 non-zero entries. Looking at Figure 2, it can be clearly noticed that there are two peaks, one in 2017 and the other in 2021. In 2017, an unexpected rise of reports for ADRs happened concerning a specific drug called Lévothyrox®. This has been marketed in France for about 40 years as a treatment for hypothyroidism and, in 2017, a new formula was introduced on the market. The Lévothyrox® case had a huge media coverage in France: Lévothyrox® spontaneous reports represent the 90% of all the spontaneous notifications that the RCPV received in 2017 (Viard et al., 2019). In addition, since the end of the year 2020, vaccinations against Covid-19 have been introduced. At that time, three vaccines are licensed in Europe, Comirnaty® was the first Covid-19 vaccine available in France in December 2020, followed by Moderna® in January 2021 and Vaxzevria® in February 2021. From Figure 2, one can understand the difficulty to work with such data which contain signals of very different amplitude. Indeed, behind those very visible effects, many ADR signals need to be detected for obvious public health reasons. In particular, those data also contain ADR reports regarding another health scandal happened in 2017, involving Mirena®, which is here far less visible than Lévothyrox®, but also led to many avoidable serious health issues.

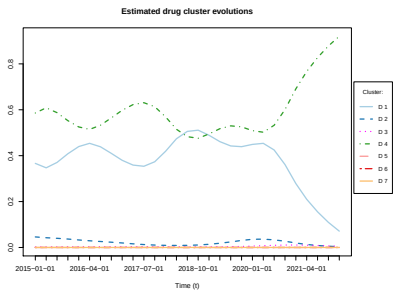

Fig. 3: Evolution of the estimates $\hat{\alpha}$.

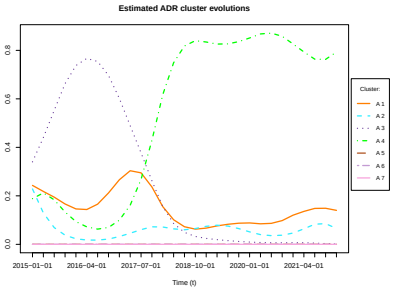

Fig. 4: Evolution of the estimates $\hat{\beta}$.

4.2 Summary of the results

To the initialize the algorithm, as explained in Section 3.4, we computed the ICL criterion on one data slice, corresponding to the first trimester, where the optimal numbers of clusters identified by the model selection criterion are $\hat{Q} = 4$ and $\hat{L} = 4$. Then, we initiated the model parameters through the cascade process described in Algorithm 2 and we ran $ZI_{\mathcal{P}}$-dLBM with $Q = 7$ and $L = 7$ to allow the model to fill or empty clusters as needed. Figure 5 depicts the estimated

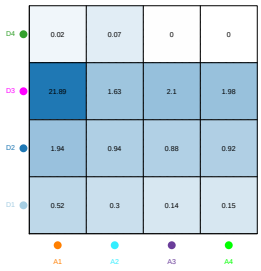

Fig. 5: Estimated Poisson intensities.

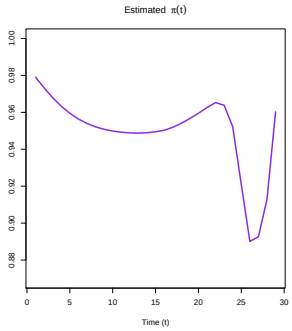

Fig. 6: Evolution of the estimates $\hat{\pi}$.

Poisson intensities Λ for $ZI_{\mathcal{P}}$-dLBM, focusing on 4 drug clusters (D) and 4 ADR clusters (A) that are activated during the inference. Each color represents a drug or ADR cluster, with higher values indicating stronger relationships (i.e., expected number of declarations received per time unit) between the respective clusters. The figure reveals varying degrees of association, for example, cluster D3 of the drug clusters is highly related with cluster A1 of ADR clusters. Figures 3, 4 and 6 show the estimates of the model parameters $\hat{\alpha}$, $\hat{\beta}$ and $\hat{\pi}$, respectively. Figure 3 shows the estimation of the mixing parameter α. Cross-referencing the

information from these results, we note that the clusters that have the highest intensity are also the less populated. For example, cluster D3 of drugs has a very high intensity of interactions with cluster A1 of adversarial effects, yet cluster D3 turns out to be very small in Figure 3. This is due to the fact that this cluster contains drugs that are declared with an an unusually high intensity. In fact, this cluster contains the drugs that are the causes of the major health crises that occurred during the reporting period: Mirena® in the first half of 2017, Lévothyrox® in the second part of 2017, and Covid-19 vaccines throughout 2021. Similarly, by analyzing the composition of cluster A1, it is possible to identify which ADRs were the most reported in each of the aforementioned crises. For instance, the most reported side effects during the Mirena® health crisis are mostly hormonal ones, such as anxiety, heat shock, and aggressive behavior. Then, looking at Figure 4, during the Lévothyrox® health crisis we notice a peak in the A1 cluster of adversarial effects, probably because the great media coverage that the scandal had in those years made people declare the most disparate side effects. Also, we see that in 2021 there is another peak, corresponding to the period of the Covid-19 vaccination. Here, the adversarial effects found in cluster A1 are mostly linked to problems related to the vaccination site (e.g. arm pain, arm inflammation, skin reaction) and flu syndrome as a result of the vaccine. Cluster D2, on the other hand, contains a few but very common and, consequently, much-reported drugs, for example, paracetamol and some of the most popular anticoagulants. From Figure 5 we note that this cluster has a stronger intensity of interactions with cluster A1 and A2 of undesirable effects. Looking at Figure 4, we note that cluster A2 is thinly populated and seems to follow the trend of health crises discussed above less closely. In fact, this cluster contains less severe and more common adversarial effects, which can occur even with the more frequent medications (e.g., itching, headache, weight gain, etc.) Clusters D1 and D4, on the other hand, are characterized by very low interaction intensities and are densely populated by all other drugs. Then, looking at Figures 5 and 4, we see that the behavior of cluster A3 of adversarial effects is very peculiar. It is characterized by almost zero interaction intensity with drug clusters D1 and D4. After the Lévothyrox® crisis, the number of reported adversarial effects significantly decreased, indicating a turning point in pharmacovigilance as people became more aware of its importance and started reporting side effects more frequently. Moreover, analysing its composition, it was noticed that at the beginning of the period it also contained all the specific side effects of Covid-19 vaccines, which were not yet known. Later, in 2021, those side effects, changed clusters moving to cluster A1 as previously described. On the other hand, Figure 6 shows the estimated evolution of the sparsity parameter over time. We see that, at the beginning of the period, in 2015, the sparsity is at 98%, then as we approach the 2017 peak, the number of declarations increases and consequently the sparsity decreases. In 2019, it again increases slightly (97%) and then decreases as we approach the peak due to the Covid-19 vaccines. In fact, at the beginning of 2021 the sparsity level reaches its minimum at a level

of 90%. Therefore, from the large initial data matrix, $ZI_{\mathcal{P}}$-dLBM was able to identify meaningful clusters of such data.

4.3 Benchmark on real data

This section focuses on comparing $ZI_{\mathcal{P}}$-dLBM with state of the art models on real-world data. We therefore carried out such an experiment by comparing $ZI_{\mathcal{P}}$-dLBM with Zip-dLBM$_{\pi(\cdot)=0}$ and dLBM discussed in appendix B.3. We also included in the comparison two models that do not consider the dynamic aspect: LBM (Robert et al., 2021), baseline for model-based co-clustering methods, and k-means (MacQueen, 1967), applied on rows and columns separately. As we are in an unsupervised context, the model performances are evaluated by the silhouette score using cosine distance on rows and columns. Table 1 displays the results of this comparison, in terms of average silhouette scores, reported with standard deviations. From the reported results, one sees that $ZI_{\mathcal{P}}$-dLBM outperforms its competitors. Also, it is worth noticing that unlike $ZI_{\mathcal{P}}$-dLBM, LBM and k-means, being independently applied at each time instant, suffer from label switching, which is not penalized in the silhouette score. This should make the interpretation of these results even more in favor for $ZI_{\mathcal{P}}$-dLBM.

	$ZI_{\mathcal{P}}$-dLBM	Zip-dLBM$_{\pi(\cdot)=0}$	dLBM	kmeans	LBM
Silhouette Score - Rows	**0.37 ± 0.12**	0.31 ± 0.12	-0.46 ± 0.25	0.21 ± 0.36	0.33 ± 0.12
Silhouette Score - Cols	**0.36 ± 0.23**	0.31 ± 0.25	-0.15 ± 0.06	0.31 ± 0.3	0.29 ± 0.23

Table 1: Results of $ZI_{\mathcal{P}}$-dLBM, Zip-dLBM$_{\pi(\cdot)=0}$, dLBM, LBM and k-means on pharmacovigilance data. Average silhouette scores are reported with standard deviations.

5 Conclusion

We have developed a dynamic co-clustering technique for simultaneously clustering rows and columns along the time dimension of a dynamic matrix. The proposed zero-inflated dynamic latent block model can be adapted to several zero-inflated probability distributions. We use a Variational EM algorithm with GD optimization to perform inference on the model's parameters, then the model is applied to a real dataset from the Regional Center of Pharmacovigilance of Nice (France) to segment drugs and adverse drug reactions based on their dynamic interactions over time. The proposed model provided a meaningful segmentation of drugs and adverse drug reactions.

Acknowledgements

This work has been supported by the French government, through the 3IA Côte d'Azur, Investment in the Future, project managed by the National Research Agency (ANR) with the reference number ANR-19-P3IA-0002.

Ethical Statement

This paper involves the analysis of pharmacovigilance data, which are obtained in collaboration with the Regional Center of Pharmacovigilance of Nice (France). It is crucial to emphasize that before the data is shared with us, a rigorous anonymization process is employed to ensure the protection of patients privacy. These anonymized data are treated as confidential and private throughout our research. We adhere to the ethical guidelines and comply with all applicable data protection regulations to safeguard the privacy and confidentiality of the individuals involved in the pharmacovigilance reporting system.

Bibliography

Ailem, M., Role, F., and Nadif, M. (2017). Sparse poisson latent block model for document clustering. *IEEE Transactions on Knowledge and Data Engineering*, 29(7):1563–1576.

Bergé, L. R., Bouveyron, C., Corneli, M., and Latouche, P. (2019). The latent topic block model for the co-clustering of textual interaction data. *Computational Statistics & Data Analysis*, 137:247–270.

Biernacki, C., Celeux, G., and Govaert, G. (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE transactions on pattern analysis and machine intelligence*, 22(7):719–725.

Bishop, C. M. (2006). Approximate inference. pages 461–517. Springer-Verlag, Berlin, Heidelberg.

Boutalbi, R., Labiod, L., and Nadif, M. (2020). Tensor latent block model for co-clustering. *International Journal of Data Science and Analytics*, pages 1–15.

Boutalbi, R., Labiod, L., and Nadif, M. (2021). Implicit consensus clustering from multiple graphs. *Data Mining and Knowledge Discovery*, 35(6):2313–2340.

Bouveyron, C., Bozzi, L., Jacques, J., and Jollois, F.-X. (2018). The functional latent block model for the co-clustering of electricity consumption curves. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 67(4):897–915.

Casa, A., Bouveyron, C., Erosheva, E., and Menardi, G. (2021). Co-clustering of time-dependent data via the shape invariant model. *Journal of Classification*, 38(3):626–649.

Corneli, M., Bouveyron, C., and Latouche, P. (2020). Co-clustering of ordinal data via latent continuous random variables and not missing at random entries. *Journal of Computational and Graphical Statistics*, pages 1–15.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22.

Gent, C. and Sheppard, C. (1992). Special feature. predicting time series by a fully connected neural network trained by back propagation. *Computing & Control Engineering Journal*, 3(3):109–112.

Govaert, G. and Nadif, M. (2003). Clustering with block mixture models. *Pattern Recognition*, 36(2):463–473.

Govaert, G. and Nadif, M. (2010). Latent block model for contingency table. *Communications in Statistics - Theory and Methods*, 39(3):416–425.

Jaakkola, T. S. and Jordan, M. I. (1997). A variational approach to bayesian logistic regression models and their extensions. In *Sixth International Workshop on Artificial Intelligence and Statistics*, pages 283–294. PMLR.

Jacques, J. and Biernacki, C. (2018). Model-based co-clustering for ordinal data. *Computational Statistics & Data Analysis*, 123:101–115.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1998). An introduction to variational methods for graphical models. In *Learning in graphical models*, pages 105–161. Springer.

Keribin, C., Brault, V., Celeux, G., and Govaert, G. (2015). Estimation and selection for the latent block model on categorical data. *Statistics and Computing*, 25(6):1201–1216.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lambert, D. (1992). Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics*, 34(1):1–14.

Li, N., Elashoff, D. A., Robbins, W. A., and Xun, L. (2011). A hierarchical zero-inflated log-normal model for skewed responses. *Statistical Methods in Medical Research*, 20(3):175–189.

Lindstrom, M. J. (1995). Self-modelling with random shift and scale parameters and a free-knot spline shape function. *Statistics in medicine*, 14(18):2009–2021.

Lomet, A. (2012). *Sélection de modèle pour la classification croisée de données continues*. PhD thesis, Compiègne.

MacQueen, J. (1967). Classification and analysis of multivariate observations. In *5th Berkeley Symp. Math. Statist. Probability*, pages 281–297.

Marchello, G., Fresse, A., Corneli, M., and Bouveyron, C. (2022). Co-clustering of evolving count matrices with the dynamic latent block model: application to pharmacovigilance. *Statistics and Computing*, 32(3):1–22.

Ospina, R. and Ferrari, S. L. (2012). A general class of zero-or-one inflated beta regression models. *Computational Statistics & Data Analysis*, 56(6):1609–1623.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in pytorch.

Ridout, M., Hinde, J., and Demétrio, C. G. (2001). A score test for testing a zero-inflated poisson regression model against zero-inflated negative binomial alternatives. *Biometrics*, 57(1):219–223.

Robert, V., Vasseur, Y., and Brault, V. (2021). Comparing high-dimensional partitions with the co-clustering adjusted rand index. *Journal of Classification*, 38(1):158–186.