The Deep Latent Position Block Model for the Clustering of Nodes in Multi-Graphs

Seydina Ousmane Niang, Charles Bouveyron, Marco Corneli, Pierre Latouche, Rémi Boutin

The Deep Latent Position Block Model for the Clustering of Nodes in Multi-Graphs

S. NIANG[1] & C. BOUVEYRON[1] & M. CORNELI[1,2] & P. LATOUCHE[3,4] & R. BOUTIN[5]

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

[2] *Université Côte d'Azur, Laboratoire CEPAM, Nice, France*

[3] *Université Clermont Auvergne, CNRS, Laboratoire LMBP, Aubière, France*

[4] *Institut Universitaire de France (IUF)*

[5] *Laboratoire de Probabilités, Statistique et Modélisation (LPSM), Sorbonne Université, Université Paris Cité, CNRS, UMR 8001*

Abstract

Network data capture relationships among actors across multiple contexts, often forming clusters of individuals. These relationships frequently involve multiple types of interactions, necessitating the use of multidimensional networks, or multigraphs, to capture their full complexity. Latent position models (LPM) embed nodes based on connection probabilities, but cannot uncover heterogeneous clusters such as disassortative patterns. Stochastic block models (SBM), in contrast, excels at clustering but lack interpretative latent representations. To address these limitations, the deep latent position block model (Deep-LPBM) was introduced to provide clustering and continuous latent space representation simultaneously in unidimensional networks . In this paper, we extend this work to multidimensional networks by introducing the deep latent position block model for multidimensional networks (Deep-LPBMM). Deep-LPBMM integrates block modeling and latent embedding across multiple interaction types, allowing nodes to partially belong to several groups, which better captures overlapping clustering structures. Our model uses a deep variational autoencoder with graph convolutional networks (GCNs) for each layer and a multi-layer perceptron to merge latent representations into a unified latent embedding representing cluster partial membership probabilities and offering effective clustering and enhanced visualization.

Keywords: Stochastic block model; Latent position model; Clustering; Multidimensional networks analysis; Multi-graphs; Variational auto encoder; Graph convolutional network

1 Introduction

Over the past decades, the study of complex networks has become increasingly relevant across various fields due to the growing abundance of structured datasets that naturally fit this framework. Networks provide a representation of connections among entities, such as individuals, organizations, or biological elements, where nodes represent the entities, and edges represent the relationships between them. In particular, when multiple types of relationships between individuals are observed across different perspectives, analyzing them simultaneously within the framework of multidimensional networks becomes crucial. Multidimensional networks have emerged as a key tool in network analysis, enabling a deeper exploration of the intricate dynamics within complex systems. Social networks are a classic example, where individuals are nodes, and their edges represent connections such as friendships or collaborations.

However, as network data grows in scale and complexity, understanding these interconnections becomes more challenging. The interdependence between nodes introduces heterogeneous connectivity patterns, making it essential to develop models that can handle both complexity and interpretability. This paper focuses on model-based approaches where observed connections are modeled probabilistically, with each node pair associated with an edge probability, indicating the likelihood of connections between dyads. These probabilities can be modeled in various ways depending on the nature of the data or the goal of the study.

1.1 Approaches for network analysis

Historical network models often relied on simple probabilistic assumptions, with edges considered as independent and edge probabilities being fixed as in Erdős and Renyi (1959) or influenced by nodes attributes (Holland and Leinhardt, 1981). Yet, real-world networks exhibit characteristics that defy these simplified approaches, such as community structures where nodes are more connected within a cluster, varying interaction patterns, and connections that depend on unobserved factors. These pioneering works allowed for the introduction of more advanced models, where the independence assumption for edges was relaxed or eliminated, and the specification of edge probabilities became more complex (Wasserman and Pattison, 1996; Robins et al., 2007; Holland, Laskey, and Leinhardt, 1983; Nowicki and Snijders, 2001; Snijders and Nowicki, 1997; Hoff, Raftery, and Handcock, 2002). Among these models, latent variable models have gained significant attention. These models assume that interconnections between nodes depend on some unobserved latent variables. The earliest latent variable models include the latent space model by Hoff, Raftery, and Handcock (2002) and the stochastic block model by Holland, Laskey, and Leinhardt (1983). Clustering nodes in graphs (Schaeffer, 2007) is one of the main exploratory tools in network analysis, providing a high-level summary of complex networks. The stochastic block model classifies nodes into groups and suppose that interactions between two nodes only depend on the clusters in which they belong to, but do not allow any direct latent representation (visualisation) of the underlying structure of the graph. Another clustering approach that allows node visualisation is the latent position cluster model (LPCM), introduced by Handcock, Raftery, and Tantrum (2007), where edge probabilities are described as a function of node positions in an unobserved latent space responsible for the observed network structure. These latent positions arise from a mixture distribution, with components corresponding to node clusters, akin to standard model-based clustering (Bouveyron et al., 2019). This approach excels at community detection since the probability of linkage increases with proximity or similarity between node positions. However, it lacks the flexibility to capture more diverse connectivity patterns. Indeed, LPCM often struggle to capture disassortative patterns, where nodes in different clusters are more likely to connect than nodes within the same cluster. For example, in hub-and-spoke structures, certain central nodes (hubs) connect widely to other nodes, but those nodes have few or no connections with each other. LPCM, which typically assumes that connection probability decreases with distance in the latent space, finds it challenging to model these cases where proximity in latent space does not necessarily imply a high connection probability. The original LPCM uses a Markov chain Monte Carlo (MCMC) algorithm for estimation and Salter-Townshend and Murphy (2013) re-implemented it using a variational Bayesian inference approach. Recent developments in this area include Boutin, Latouche, and Bouveyron (2023), which uses variational inference procedure based on deep learning and extends the deep latent position cluster model to textual data analy-

sis. Many other deep learning based models exist on the state-of-the-art of network analysis. The variational autoencoder (VAE), introduced for Euclidean data by Kingma, Welling, et al. (2019) and Rezende and Mohamed (2015), inspired Kipf and Welling (2016b) to create the variational graph auto-encoder (VGAE) for network analysis. This model combines probabilistic methods with deep neural networks to capture the underlying structure of networks. Building on VGAE, Pan et al. (2018) enhanced it by applying adversarial inference as a regularization strategy, resulting in improved performance. However, both VGAE and its variants require external clustering methods, such as k-means, on the learned node representations to achieve node clustering. To address this limitation, Mehta, Duke, and Rai (2019) introduced an adaptation of the overlapping stochastic block model (Latouche, Birmelé, and Ambroise, 2009) that leverages neural networks to encode node embeddings. More recently, Liang et al. (2022) developed the deep latent position model (Deep-LPM), which integrates probabilistic modeling and neural embeddings in a unified framework. However, Deep-LPM, similar to the latent space model (LSM), assumes assortative structures for edge probabilities, limiting its ability to analyze disassortative networks.

One limitation of the clustering methods described above is the assumption that each node belongs to a single group. In reality, individuals often belong to multiple overlapping groups, reflecting the multiple roles they play within a social context. Consider, for example, a researcher who specializes in both biology and computer science. It would be reasonable to represent this individual as partly belonging to two different fields or clusters, as his expertise overlaps both disciplines. The mixed membership stochastic block (MMSBM, Airoldi et al., 2006) model was proposed for this purpose. To account for such partial memberships, the bayesian partial membership model (BPM; Heller, Williamson, and Ghahramani, 2008) was proposed for classical variable data matrix, offering a way for nodes to belong partially to different clusters. BPM makes use of latent continuous variables modeling the probabilities of each data-point to belong to the different clusters. Recently, Boutin, Latouche, and Bouveyron (2024) introduced the deep latent position block model (Deep-LPBM), a novel approach combining the strengths of SBM, LPCM, BPM and MMSBM. Deep-LPBM enables both visualization within a latent space and partial membership block modeling, providing a more versatile clustering framework than traditional clustering methods, with the ability to capture disassortative and hub patterns, to handle overlapping groups and to visualize the underlying network structure in a latent space.

1.2 Approaches for the analysis of multi-graphs

In many real-world scenarios, entities are interconnected through a variety of relationships and can participate in multiple forms of interactions, resulting in complex, multi-layered structures known as multidimensional networks or multi-graphs. Such structures arise when distinct types of connections are recorded among the same set of nodes, capturing the multifaceted nature of social, biological, or technological interactions. For example, within a professional context such as a research institution, individuals may connect on multiple levels. These interactions could be professional, such as collaborations on research projects, mentorship, or seeking and providing expert advice, and personal, such as forming friendships, participating in recreational activities, or sharing social events. In each case, the same set of nodes (the individuals) interact through multiple channels or dimensions, forming a rich and layered web of relationships that cannot be accurately represented within a single, unidimensional framework.

Consequently, multidimensional networks capture this diversity by encompassing various types of links or edges across different views or layers of the network, with each view representing a particular type of relationship. Analyzing these networks requires approaches that can simultaneously model each layer and capture both within-layer and cross-layer structural patterns. Traditional single-layer network analysis methods fall short when applied to multidimensional data because they miss the nuanced dependencies between different interaction types. However, many of them have been extended to the multidimensional framework.
For latent position models, a significant extension is the latent space joint model (LSJM) introduced by Gollini and Murphy (2016). This model extends the latent space model (LSM) to accommodate multiple network views by assuming that the probability of connections between nodes across these views is determined by a shared latent variable. Each layer of the network has its own latent space, but these are treated as instances of an overarching latent space that represents the average latent positions of nodes, which in turn influences connectivity across all layers. LSJM is estimated through an expectation-maximization (EM) algorithm, where each layer is independently fitted to produce initial parameter estimates. These initial estimates are refined to update the joint posterior distribution of the LSJM, iteratively improving until convergence. However, this method faces computational constraints, especially with networks that have a large number of nodes or layers. D'Angelo, Murphy, and Alfò (2019) adapted the LSJM framework specifically for studying the Euro-Vision dataset. They introduced network-specific coefficients to weight the role of the latent space in determining edge probabilities in each network, improving the model flexibility across different layers.
Several studies have introduced clustering methods based on latent space models, extending LPCMs to multidimensional networks. These methods use multiple views to construct the latent space, which is then employed for clustering assignments. Among them, we can cite D'Angelo, Alfò, and Fop (2023), where the authors propose an infinite mixture latent position cluster model for single and multidimensional network data (IM-LPCM). The proposed framework allows to jointly estimate cluster parameters and latent coordinates without previous specification of the number of clusters. This model assumes that the latent coordinates arise from an infinite mixture of Gaussian distributions allowing to treat the number of mixture components, and consequently the number of clusters, as a model parameter, on which inference is performed. IM-LPCM is estimated within a hierarchical Bayesian framework and inference is carried out using a MCMC algorithm. This approach also suffers from computational issues and struggles capturing structures more complex than communities as it is the case for the original LPCM. Another extension of LPCM include those by Sewell and Chen (2017) for clustering longitudinal network data, a special case of multi-graphs where multiple views represent a single social relation recorded at different times.

Within the block modeling framework, several extensions of SBM have been proposed. A key model that applies block modeling to multi-graphs is the multilayer stochastic block model (MLSBM) introduced by Holland, Laskey, and Leinhardt (1983). This model groups nodes into classes, forming blocks within the multi-graph, with the assumption that nodes maintain the same block structure across layers, even if connection probabilities vary between layers. The single-layer version of this model is the SBM, which MLSBM directly extends. Han, Xu, and Airoldi (2015) investigated the asymptotic properties of spectral clustering and maximum-likelihood estimation (MLE) within this framework, proposing a computationally feasible variational approximation for the MLE in large networks. Paul and Chen (2016) further explored

MLE consistency when node or edge types increase simultaneously. Stanley et al. (2016) introduced one of the first approaches that integrated a multilayer SBM with a mixture of layers, using a two-step greedy inference method. In the initial step, it infers a SBM for each layer and groups together SBMs with similar parameters. In the second step, these outcomes serve as the starting point for an iterative procedure that simultaneously identifies the strata spanning the layers. In each stratum, the nodes are independently distributed into blocks. Rebafka (2024) proposed a Bayesian framework for a finite mixture of MLSBM and employed a hierarchical agglomerative algorithm for the clustering process. It initiates with individual singleton clusters and then progressively merges clusters of networks according to an integrated classification likelihood (ICL Biernacki, Celeux, and Govaert, 2000) criterion also used for model selection. Barbillon et al. (2017) proposed the multiplex stochastic block model (MSBM), an extension of the MLSBM. Like the methods mentioned earlier, MSBM assumes that in each view the relationships between any two individuals are independent of those involving other individuals. However, it differs by accounting for dependencies between different types of relationships among individuals (interdependence between views). This results in a highly complex model, where the number of parameters increases exponentially with the number of layers and polynomial with the number of clusters. The model parameters are estimated using an extended variational EM (VEM) algorithm, and the optimal number of blocks is determined using an ICL penalized likelihood criterion.
In this same field, De Santiago, Szafranski, and Ambroise (2024) recently proposed the mixture of multilayer integrator stochastic blocks model (mimi-SBM), a Bayesian mixture of multilayer SBM that takes into account several sources of information, and aggregates the different partitions found thank to the framework of meta consensus clustering (Monti et al., 2003; Fred and Jain, 2005) which is a technique used to find a single partition from multiple clustering solutions. It uses a mixture model to cluster the views with the aim to deal with the redundant and complementary information sources in order to draw the maximum of information from the different views. A Bayesian selection criterion is developed for the selection of the number of views clusters and nodes clusters.

In this work, we introduce the deep latent position block model for multidimensional networks (Deep-LPBMM). This framework is built within the class of latent position block models, extending the work of Boutin, Latouche, and Bouveyron (2024) and benefiting from the advantages of both stochastic block models and latent space models by assuming that block partial membership depends on the same latent variables across all different views. It allows for the joint estimation of clustering allocations and latent coordinates. Furthermore, the framework supports partial membership, enabling nodes to belong to multiple groups simultaneously, which captures the nuanced and overlapping structures commonly observed in real-world networks.

1.3 Main contributions and organisation of the paper

- This paper extends the work of Boutin, Latouche, and Bouveyron (2024) to the clustering of nodes in multidimensional networks. It is capable of i) analyzing multidimensional networks using a variational graph autoencoder approach, ii) providing a visualisation of the entire network compatible with block modeling and iii) performing block modeling, as well as node partial memberships estimation.

- Deep-LPBMM is able to associate each node with several connectivity patterns, rendering

refined results as illustrated in the analysis of the real world dataset.

- Deep-LPBMM proposes for each view the same block decoder as in Deep-LPBM and combines GCN based encoders and a multi-layer perceptron to produce an unified encoder, adapted to multidimensional networks, to model any type of connectivity patterns

The rest this paper is structured as follows: in Section 2, we introduce the latent position block model for clustering nodes within multi-graphs, outlining its key components and parameter estimation strategy. Section 3 details the key aspects of the Monte Carlo variational algorithm used for inference. A simulation study evaluating the model's capacity in recovering the latent space and node clustering structure is provided. Moreover an introductory example is given in Section 4. Then, Section 5 includes an illustrative example involving the analysis of professional and personal relationships among employees in an institutions. This example examines a multi-graph representing interactions among attorneys at a law firm. Finally, a conclusion is given in Section 6.

2 The latent position block model for the clustering of nodes in multi-graphs

We consider a multi-graph $\mathcal{G}$ characterised by a set of L graphs defined on the same set of nodes $\mathcal{V}$, whose cardinality is N. We identify a graph with its adjacency matrix A, a square $N \times N$ matrix such that its element A_{ij} is 1 when there is an edge from vertex i to vertex j, and 0 otherwise. So, our multi-graph $\mathcal{G}$ is the set $\{A^{(\ell)}\}_{\ell=1}^{L}$ where $A^{(\ell)}$ represents the ℓ-th graph. The n-dimensional simplex is denoted $\Delta_n = \{p \in \mathbb{R}^n\colon \forall i \quad p_i \geq 0 \text{ and } \sum_{i=1}^{n} p_i = 1\}$.
In the following, the views are assumed to be undirected, meaning that each $A^{(\ell)}$ is a symmetric matrix.

2.1 Generative model for multi-graphs

In this section, we present the modelling assumptions in Deep-LPBMM. In the same vein as in Gollini and Murphy, 2016, we suppose that the edge probabilities in the different views are explained by a single latent variable. This implies that all views, despite their potentially different nature, are unified under a common latent representation that reflects the underlying structural patterns of the network. In more details, the N nodes are assumed to belong to Q clusters in a non exclusive manner.
Assuming for the moment that the number of clusters Q is fixed, each node $i \in \{1, \dots N\}$ is assumed to have a single latent representation Z_i drawn from a centred and reduced Gaussian distribution:

$$Z_i \overset{\text{iid}}{\sim} \mathcal{N}_d(0, \mathbb{I}_d),$$

with $d = Q - 1$.
The set of node embeddings is denoted by $Z := \{Z_i\}_i$ in the rest of the paper.
As in Boutin, Latouche, and Bouveyron (2024), to link the latent representation of the nodes Z with block modeling, we rely on the bijective softmax transformation of the latent positions

$h : \mathbb{R}^{Q-1} \to \Delta_Q$ such that:

$$\eta_{iq} := h(Z_i) := \begin{cases} \frac{\exp(Z_{iq})}{1+\sum_{r=1}^{Q-1}\exp(Z_{ir})} & \text{if} \quad q \neq Q, \\ \\ \frac{1}{1+\sum_{r=1}^{Q-1}\exp(Z_{ir})} & \text{if} \quad q = Q, \end{cases} \tag{1}$$

and we denote $\eta = (\eta_1, \ldots, \eta_N)^\intercal$. The mapping h aims at encoding Z into cluster partial membership probabilities. Thus η_{iq} is the probability that node i is in cluster q. Eventually, we suppose that the probability of connection between two nodes follows a Bernoulli distribution with parameters depending on η such that:

$$\begin{aligned} A_{ij}^{(\ell)} | \eta_i, \eta_j &\overset{\text{i.i.d.}}{\sim} \text{Bernoulli}(\eta_i^\intercal \Pi^{(\ell)} \eta_j), \\ p(A^{(\ell)} | \eta, \Pi^{(\ell)}) &= \prod_{i<j} (\eta_i^\intercal \Pi^{(\ell)} \eta_j)^{A_{ij}^{(\ell)}} \cdot \left(1 - \eta_i^\intercal \Pi^{(\ell)} \eta_j\right)^{1-A_{ij}^{(\ell)}}, \end{aligned} \tag{2}$$

where the $Q \times Q$ matrix $\Pi^{(\ell)} = (\Pi_{qr}^{(\ell)})_{1 \leq q,r \leq Q}$ is symmetric and refers to the connectivity matrix whose entry (q, r) is the probability that a node in block q is connected to a node in block r in the ℓ-th view. We also used the simplified notation $\prod_{i<j} = \prod_{i=1}^{N-1} \prod_{j=i+1}^{N}$. Here we suppose that the probability of two nodes to be linked may change across views depending on the connectivity matrix of the view under question.

The work of Boutin, Latouche, and Bouveyron (2024) was proposed for single view networks in order to estimate simultaneously clusters of nodes as well as a visualisation (Z) of the network. Moreover, unlike standard block modeling (SBM Snijders and Nowicki, 1997), this work allows to capture partial membership of nodes to clusters (Heller, Williamson, and Ghahramani, 2008). Partial membership allows the observations (nodes in our case) to play several roles in multiples clusters.
We further suppose the independence between different views given Z, resulting in the following joint distribution of (A, Z):

$$p(A, Z | \Pi) = \prod_{l=1}^{L} p(A^{(\ell)} | Z, \Pi^{(\ell)}) p(Z).$$

The graphical model of Deep-LPBMM is given in Figure 1.

2.2 Link with other models

This paper extends the work of Boutin, Latouche, and Bouveyron (2024) who clearly established links with stochastic block model (SBM Holland, Laskey, and Leinhardt, 1983), mixed-membership stochastic block model (Heller, Williamson, and Ghahramani, 2008) and latent position model (Hoff, Raftery, and Handcock, 2002) in the case of single layered graphs.

Our model is also directly related to the multi-graph stochastic block model (Holland, Laskey, and Leinhardt, 1983) which assumes a discrete latent variable η coming from a multinational distribution ($\eta \sim \mathcal{M}(1, \pi = \pi_{1:Q})$) instead of a logistic softmax distribution in our

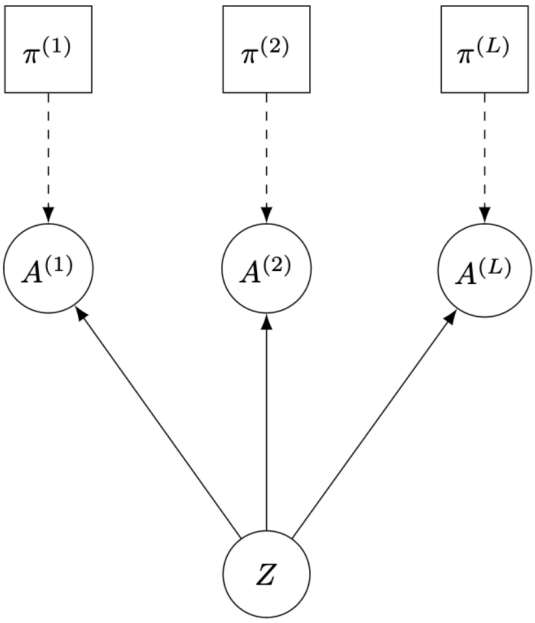


Figure 1: Graphical model of Deep-LPBMM. $A^{(\ell)}$ represents the adjacency matrix for the ℓ-th view. The probability of a link between nodes i and j may vary across views, depending on $\Pi^{(\ell)}$. The matrix $Z \in \mathbb{R}^{N\times(Q-1)}$ encodes information about the likelihood of individuals belonging to different clusters.

case. Our choice allows the nodes to partially belong to several clusters and provide us with node embeddings in a latent space of dimension $Q-1$.

We also establish the connection between Deep-LPBMM and the multiplex stochastic block model (MSBM, Barbillon et al., 2017). Indeed given η, MSBM does not assume conditional independence of the edges at different layers. In more details, for any nodes pair (i,j), in MSBM one has $p(A_{ij}^{1:L}|\eta_i,\eta_j) \neq \prod_{\ell=1}^{L} p(A_{ij}^{\ell})|\eta_i,\eta_j)$. Our model can be obtained from MSBM by assuming that conditional independence and modeling η_i via 1 instead of viewing it as a multinomial random variable.

3 Inference and optimisation

In order to estimate the model parameters $\Pi := \{\Pi^{(\ell)}\}^{\ell}$, we rely on the marginal log-likelihood of the multi-graph, after integrating away the latent variable Z

$$\log p(A|\Pi) = \log \int_Z p(A, Z|\Pi) dZ.$$

Unfortunately, because of the softmax function, this quantity is hard to compute. In addition, an expectation-maximisation (EM) algorithm cannot be employed directly since the posterior distribution $p(Z|A, \Pi)$ is not tractable. Consequently, we choose to rely on a variational inference strategy for approximation purposes.

3.1 Variational decomposition of the marginal log-likelihood

For any distribution $q(\cdot)$ for the latent variable Z, in force of the Jensen's inequality, the following decomposition holds:

$$\begin{aligned}
\log p(A|\Pi) &= \log \int_Z \frac{p(A, Z|\Pi)}{q(Z)} q(Z) dZ \\
&= \log \mathbb{E}_{q(Z|A)} \frac{p(A, Z|\Pi)}{q(Z|A)} \\
&\geq \mathbb{E}_{q(Z)} \left[\log \frac{p(A, Z|\Pi)}{q(Z)} \right] \\
&:= \mathcal{L}(\Pi, q(\cdot)).
\end{aligned}$$

In this paper, we refer to $\mathcal{L}(\Pi, q(\cdot))$ as the evidence lower bound (ELBO). Furthermore, the exact difference between $\log p(A|\Pi)$ and $\mathcal{L}(\Pi, q(\cdot))$ is

$$\mathrm{KL}(q(\cdot)||p(Z|A, \Pi)) = \log p(A|\Pi) - \mathcal{L}(\Pi, q(\cdot)), \tag{3}$$

where the Kullback-Leibler (KL) divergence between q and the posterior distribution $p(Z|A, \Pi)$ is always non-negative, indicating that the ELBO is a lower bound of the marginal log-likelihood. Since the marginal log-likelihood does not depend on $q(\cdot)$, maximizing the ELBO with respect to $q(\cdot)$ is equivalent to minimizing the Kullback-Leibler divergence between $q(\cdot)$ and the posterior distribution. Since this minimisation is satisfied when the variational $q(\cdot)$ is equal to the true posterior distribution $p(Z|A, \Pi)$ which is not tractable here, we restrict the family of variational distributions by assuming a mean-field approximation and other hypotheses to make the ELBO tractable:

$$q(Z) := q(Z|A) = \prod_{i=1}^{N} q_\phi(Z_i|A) = \prod_{i=1}^{N} \mathcal{N}(Z_i; \bar{\mu}_{\phi,\xi}(A)_i, \bar{\sigma}^2_{\phi,\omega}(A)_i \mathbb{I}_{Q-1}), \tag{4}$$

with

$$\begin{aligned}
\phi &= (\phi^{(1)}, \dots \phi^{(L)}), \\
\bar{\mu}_{\phi,\xi}(A) &= \mathrm{MLP}_\xi(\mu^{(1)}_{\phi^{(1)}}, \dots, \mu^{(L)}_{\phi^{(L)}}), \\
\bar{\sigma}_{\phi,\omega}(A) &= \mathrm{MLP}_\omega(\sigma^{(1)}_{\phi}, \dots, \sigma^{(L)}_{\phi}),
\end{aligned}$$

$$\left[\mu^{(\ell)}_{\phi^{(\ell)}}, \sigma^{(\ell)}_{\phi^{(\ell)}}\right] = f^{(\ell)}_{\phi^{(\ell)}}(\bar{A}^{(\ell)}),$$

where $f^{(l)}_{\phi^{(\ell)}} : \mathbb{R}^{N\times N} \to \mathbb{R}^{K+1}$ is a graph convolutional network (GCN, Kipf and Welling, 2016a) mapping the normalized adjacency matrix $\bar{A}^{(\ell)} = \tilde{D}^{(\ell)^{-\frac{1}{2}}}(A^{(\ell)} + I_N)\tilde{D}^{(\ell)^{-\frac{1}{2}}}$ into the vector of the variational means and the log standard deviation. K denotes the dimension of the intermediate latent space for the variational means.$\tilde{D}^{(\ell)}$ represents the diagonal matrix corresponding to the degree of nodes with respect to the view $(A^{(\ell)} + I_N)$ defined as $\tilde{D}^{(\ell)} = 1 + \sum_{i=1}^{N} A^{(\ell)}_{ij}$. Regarding the encoder of the adjacency matrix, we based our neural network architecture on Kipf and Welling (2016b). $\text{MLP}_\xi : \mathbb{R}^{L\times K} \to \mathbb{R}^{Q-1}$ and $\text{MLP}_\omega : \mathbb{R}^{L} \to \mathbb{R}$ denote multi-layer perceptrons that allow us to aggregate the variational parameters from the different views. In Equation 4, $\bar{\mu}_{\phi,\mathbf{x}}(A)_i$ denotes the i-th row of $\bar{\mu}_{\phi,\mathbf{x}}(A)$, and $\bar{\sigma}^2_{\phi,\omega}(A)_i$ the i-th element of $\bar{\sigma}^2_{\phi,\omega}(A)$.

Thus, the ELBO can be decomposed as follows:

$$\begin{aligned}
\mathcal{L}(q,\Pi) &= \mathbb{E}_{q(\cdot)}\left[\log \frac{p(A,Z|\Pi)}{q(Z|A)}\right] \\
&= \mathbb{E}_{q(\cdot)}\left[\log \frac{\prod_{\ell=1}^{L} p(A^{(\ell)}|Z,\Pi)p(Z)}{q(Z|A)}\right] \\
&= \mathbb{E}_{q(\cdot)}\left[\sum_{l} \log p(A^{(\ell)}|Z,\Pi) + \log p(Z) - \log q(Z|A)\right] \\
&= \mathbb{E}_{q(\cdot)}\left[\sum_{l} \log p(A^{(\ell)}|Z,\Pi)\right] - \mathbb{E}_{q(\cdot)}\left[\log \frac{q(Z|A)}{p(Z)}\right] \\
&= \underbrace{\mathbb{E}_{q(\cdot)}\left[\sum_{l} \log p(A^{(\ell)}|Z,\Pi)\right]}_{\text{Reconstruction term}} - \underbrace{\text{KL}(q(Z|A)||p(Z))}_{\text{Regularisation term}}.
\end{aligned}$$

Finally:

$$\begin{aligned}
\mathcal{L}(q,\Pi) =& \mathbb{E}_{q(\cdot)}\left[\sum_{\ell=1}^{L}\sum_{i<j}^{N}\left(A^{(\ell)}_{ij}\log \eta_i^\intercal \Pi^{(\ell)}\eta_j + (1-A^{(\ell)}_{ij})\log(1-\eta_i^\intercal\Pi^{(\ell)}\eta_j)\right)\right] \\
&- \sum_i \left[-(Q-1)\log\bar{\sigma}_{\phi,\omega}(A)_i - \frac{Q-1}{2} + \frac{1}{2}||\bar{\mu}_{\phi,\xi}(A)_i||_2^2 + \frac{Q-1}{2}\bar{\sigma}_{\phi,\omega}(A)_i^2\right].
\end{aligned} \tag{5}$$

The regularization term, which involves the KL divergence between q and the posterior, can be computed exactly. The main challenge lies in calculating the reconstruction term. To address this and optimize the ELBO, we introduce a stochastic gradient descent algorithm, detailed in the next section

3.2 Monte Carlo variational algorithm

This section details the optimisation algorithm as well as the initialisation step and a model selection criterion introduced in order to select the value of Q.

The model parameters Π, and variational parameters ϕ, ξ and ω cannot be updated with analytical formulas because of the integral involving the variational distribution $q(\cdot)$ in the ELBO. In this section, we aim at deriving estimates $\tilde{\mathcal{L}}(\cdot)$ of the ELBO $\mathcal{L}(\cdot)$ to perform stochastic gradient descent:

$$\begin{aligned}\tilde{\mathcal{L}}(q, \Pi) :=& \frac{1}{S}\sum_{s=1}^{S}\left[\sum_{\ell=1}^{L}\sum_{i<j}^{N}\left(A_{ij}^{(\ell)}\log \eta_i^{\intercal(s)}\Pi^{(\ell)}\eta_j^{(s)} + (1-A_{ij}^{(\ell)})\log(1-\eta_i^{\intercal(s)}\Pi^{(\ell)}\eta_j^{(s)})\right)\right] \\ &- \sum_i\left[-(Q-1)\log\bar{\sigma}_\phi(A)_i - \frac{Q-1}{2} + \frac{1}{2}||\bar{\mu}_\phi(A)_i||_2^2 + \frac{Q-1}{2}\sigma_\phi(A)_i^2\right],\end{aligned} \tag{6}$$

where $\eta_i^{(s)} = h(Z_i^{(s)})$ and $Z_i^{(s)} \sim q(\cdot)$. In order to make the variational parameters appear in the first term on th right hand side of the above equality, we adopt the reparameterization trick (Kipf and Welling, 2016b). In particular, if $\epsilon^{(s)} \sim \mathcal{N}(0, \mathbb{I}_{Q-1})$, then $Z_i^{(s)} = \bar{\mu}_{\phi,\xi}(A)_i + \bar{\sigma}^2_{\phi,\omega}(A)_i\ \epsilon^{(s)} \sim \mathcal{N}(\bar{\mu}_{\phi,\xi}(A)_i, \bar{\sigma}^2_{\phi,\omega}(A)_i\mathbb{I}_{Q-1}) = q(\cdot|A)$.
To apply a gradient descent algorithm as done in Boutin, Latouche, and Bouveyron (2024), we map the constrained values $(\Pi_{qr}^{(\ell)})_{q,r}$ from the interval $]0,1[$ to the unconstrained set $\mathbb{R}$ using a function g defined as:

$$g : \begin{cases}\mathbb{R} \to]0,1[\\ x \longmapsto 0.5 + \frac{1}{\pi}\arctan(x).\end{cases}$$

This function g is bijective, mapping each real number to a unique value in $]0,1[$. Letting $\Pi^{(\ell)} = g((\tilde{\Pi}^{(\ell)})$, we can optimize the ELBO w.r.t. $\tilde{\Pi}^{(\ell)} = (\tilde{\Pi}_{qr}^{(\ell)})_{q,r}$, an unconstrained $Q \times Q$ matrix, using gradient descent. For clarity, we denote $\Pi^{(\ell)} = g(\tilde{\Pi}^{(\ell)})$ as the element-wise mapping of $\tilde{\Pi}^{(\ell)}$ by g.

Initialisation of the parameters

An effective initialization is essential for algorithms such as expectation-maximization (EM) or variational EM (V-EM). Due to the non-convex nature of the related optimization problems, there is a risk of getting trapped in local stationary points, which can lead to sub-optimal solutions. In order to deal with this issue, we test several initializations for both the model and the variational parameters and select the final estimates associated with the highest ELBO. We first select a view, denoted as view k, and follow these steps:

1) **Initialize η via K-means clustering**: Using the k-th view, apply K-means clustering to the rows of $A^{(k)}$ and use the resulting clusters to initialize η via Equation 1.

2) **Compute initial latent positions**: Derive the initial latent positions as $\hat{Z}^0 = \text{Softmax}_{\text{inv}}(\eta)$, where $\text{Softmax}_{\text{inv}}$ represents the inverse of the bijective softmax function $h(\cdot)$ defined in Equation 1.

3) **Optimize the encoding of the latent positions and variance**: Encode all the views and optimize the following objective with respect to ϕ, ω, and ξ using stochastic gradient descent:

$$\sum_{i=1}^{N}\frac{1}{N}\left\{\left|\left|\bar{\mu}_{\phi,\xi}(A)_i - \hat{Z}_i^0\right|\right|_2^2 + \left|\left|\bar{\sigma}^2_{\phi,\omega}(A)_i - 0.01\right|\right|_2^2\right\}.$$

4) **Initialize block connectivity parameters**: Set the initial values of $\Pi_{qr}^{(\ell)}$ as $\hat{\Pi}_{qr}^{(\ell)} = \frac{N_{qr}}{N_q N_r}$, where N_{qr} denotes the number of edges between block q and block r, and $(N_q)_q$ is a vector whose entry q denotes the size of the q-th cluster identified by K-means.

In addition to the L views, we explore two composite approaches using the concatenation and the sum of all views, respectively labeled as views $L+1$ and $L+2$, so we introduce the $N \times N \times L$ matrix $A^{(L+1)} = [A^{(1)}, \dots A^{(L)}]$ and $A^{(L+2)} = A^{(1)} + \cdots + A^{(L)}$ and repeat the steps 1) to 4) described above. These combined view serves as alternative initialization options that may capture complementary or overlapping information from multiple views. Thus, we compare $L+2$ initialization processes and select the one that produces the highest ELBO after optimizing the model. The optimisation procedure and the initialisation are respectively summarised in Algorithms 1 and 2.

Algorithm 1: Initialisation algorithm

Data: $A = \{A^{(\ell)}\}^{\ell}$, $l^{\star}$

Initialisation step:

Compute $\eta_{\text{init}} = \text{Onehot}(\text{KMEANS}(A^{(\ell^{\star})}))$;

Compute $\hat{Z}^0 = \text{Softmax}_{\text{inv}}(\eta_{\text{init}})$;

Compute $\widehat{\Pi}^{(\ell)} = \frac{N_{qr}^{(\ell)}}{N_q^{(\ell)} N_r^{(\ell)}}$, where $N_{qr}^{(\ell)}$ represents the number of edges between blocks q and r, and $N_q^{(\ell)}$ and $N_r^{(\ell)}$ represent, respectively, the sizes of clusters q and r ;

while *not converged* **do**

Take gradient step on

$$\nabla_{\xi,\phi,\omega} \sum_{i=1}^{N} \frac{1}{N} \left\{ \left\| \bar{\mu}_{\phi,\xi}(A)_i - \hat{Z}_i^0 \right\|_2^2 + \left\| \bar{\sigma}_{\phi,\omega}^2(A)_i - 0.01 \right\|_2^2 \right\}$$

if *converged* **then**

Break ;

Return $\hat{\Pi}, \hat{\phi}, \hat{\xi}, \hat{\omega}$;

3.2.1 Model selection

So far, we assumed that the number of cluster was given. In this section, we discuss how to select a value of Q that maximize the informative value captured from the data while avoiding over-parameterization. Following the approach recommended by Boutin, Latouche, and Bouveyron (2024), we adopt the Akaike Information Criterion (AIC) (Akaike, 1974) as the primary model selection metric. AIC serves as an estimator of prediction error, providing an assessment of each model relative quality based on a given dataset. When comparing a collection of models, AIC evaluates the trade-off between model fit and complexity, allowing for an informed choice of model that best balances accuracy and generalizability. This criterion thus offers a robust framework for model selection by prioritizing the model that most effectively captures the underlying structure without undue complexity.

Algorithm 2: Variational algorithm

Data: $A = \{A^{(\ell)}\}^{\ell}, \hat{\Pi}, \hat{\phi}, \hat{\xi}, \hat{\omega}$
Estimation of Deep-LPBMM:
for *epoch* $\in \{1, \ldots, max\ iter\}$ **do**
  Compute $\mu = \bar{\mu}_{\phi,\xi}(A)$ and $\sigma = \bar{\sigma}^2_{\phi,\omega}(A)$;
  Sample $\epsilon \sim \mathcal{N}(0, I)$;
  Update $\hat{Z} \leftarrow \mu + \sigma\epsilon$;
  Compute $\hat{\eta} = \text{Softmax}(\hat{Z})$;
  Compute $l(\hat{\Pi}, \phi, \omega, \xi) \leftarrow$ Plug $\hat{\eta}, \mu, \sigma$ into Equation 6 ;
  Perform stochastic gradient descent on $l(\cdot)$ w.r.t. $\hat{\Pi}, \omega, \xi, \phi$;
Return optimal parameters and corresponding latent variables ;

$$AIC(Q) := \sum_{l=1}^{L} \left(\log p(A^{(\ell)} | \hat{Z}, \hat{\Pi}^{(\ell)}) - \frac{Q(Q+1)}{2} \right) - N(Q-1)$$

where $\hat{\Pi}^{(\ell)}$ denotes the estimated $\Pi^{(\ell)}$ after maximizing the ELBO and $\hat{Z} = \bar{\mu}_{\phi,\xi}(A)$ with ϕ and ξ the variational estimates after optimizing the ELBO.
To select the best number of cluster Q for a given multidimensional network, we fitted the Deep-LPBMM model with several candidate values for Q and selected the value Q that gave the highest AIC. The relevance of this criterion is assessed in the next section on synthetic data.

4 Evaluation on synthetic datasets

In order to evaluate the relevance of the network representations and the node partitions obtained through our methodology, we compare them with the ground truth on synthetic data. First, we present the network structures and simulation scenarios used in this section and the evaluation strategy of the method we propose. Second, we introduce an example of Deep-LPBMM usage and finally we compare the performances of our approach with state-of-the-art methods in some challenging scenarios where models designed for single-layer networks may struggle.

4.1 Network structures, simulated scenarios and evaluation

To evaluate the ability of our method to represent diverse network topologies, we tested it on three different view structures, each consisting of 200 nodes. For all views, we assumed stochastic equivalence among nodes within the same cluster. This means that the likelihood of a link between two nodes depends solely on the clusters to which they belong. This assumption aligns with the stochastic block model (SBM), resulting in the formation of a connectivity matrix Π, with its values determined by the specific structure under study.

Network structures For the connectivity matrix Π, we considered three different network structures each reflecting a kind of social interaction: i) the **community structure**, where we assume that nodes in the same group have a high probability of connection indicated by β and that nodes in different groups have a low probability of connection indicated by δ, ii) the **disassortative structure** where we assume that nodes in different clusters have a high probability of connection denoted by β, and that nodes in the same cluster have a low probability of connection denoted by δ, iii) a **hub structure** where we assume that one of the clusters is highly connected to all the clusters, with a probability β and that the other clusters are communities.

- Community structure

$$\Pi = \begin{pmatrix} \beta & \delta & \dots & \delta \\ \delta & \beta & \dots & \delta \\ \dots & \dots & \dots & \dots \\ \delta & \delta & \dots & \beta \end{pmatrix} = (\beta - \delta)\mathbb{I}_Q + \delta \mathbb{1}_Q$$

  with $\beta > \delta$.

- Disassortative structure

$$\Pi = \begin{pmatrix} \delta & \beta & \dots & \beta \\ \beta & \delta & \dots & \beta \\ \beta & \dots & \dots & \beta \\ \beta & \beta & \dots & \delta \end{pmatrix} = (\delta - \beta)\mathbb{I}_Q + \beta \mathbb{1}_Q$$

  with $\beta > \delta$.

- Hub structure

$$\Pi = \begin{pmatrix} \beta & \beta & \beta & \dots & \beta \\ \beta & \beta & \delta & \dots & \delta \\ \dots & \dots & \dots & \dots & \dots \\ \beta & \delta & \delta & \dots & \beta \end{pmatrix}$$

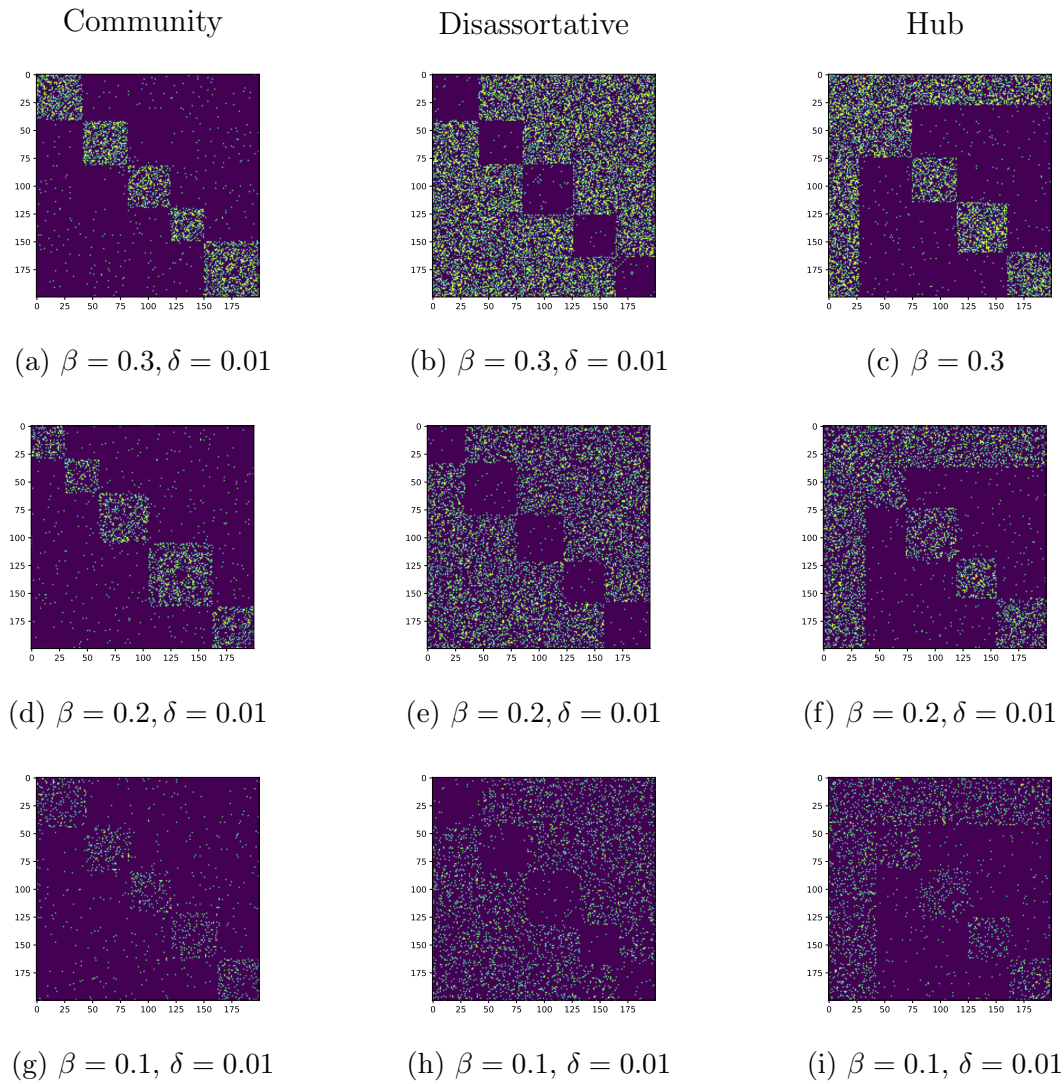
  with $\beta > \delta$.

Here $\mathbb{I}_Q$ refers to the identity matrix and $\mathbb{1}_Q$ is such that $\forall i, j \leq Q \ \mathbb{1}_{Qij} = 1$. In Figure 2, we plot some adjacency matrices, reorganized by blocs, of the three different structures presented above with different connectivity levels β.

To build multidimensional networks from these three network structures, we designed three different simulation scenarios.

Simulated scenarios

- **Scenario 1**: to highlight the usefulness of multi-view clustering over single-layer clustering, we start with a scenario where two views of the same network bring complementary information about the optimal partition of the nodes, making it impossible to recover these clusters by examining just one of the network views.
  More specifically, the first view consists of 200 individuals divided into two communities.



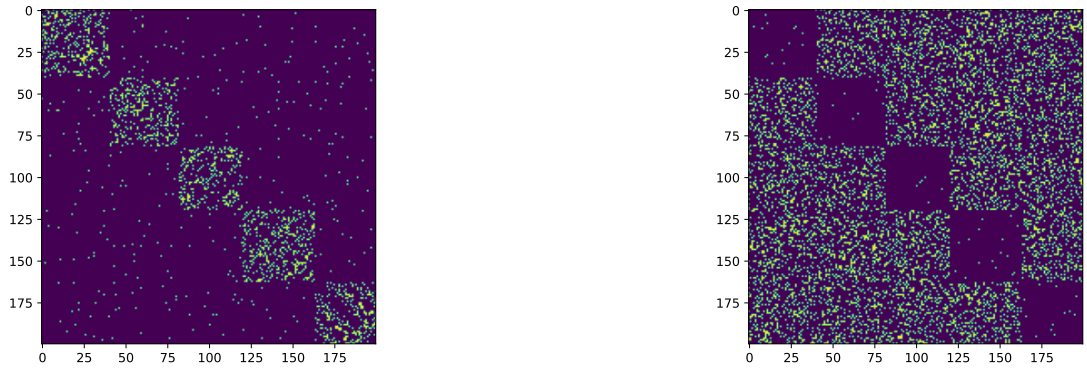
(a) $\beta = 0.3, \delta = 0.01$ (b) $\beta = 0.3, \delta = 0.01$ (c) $\beta = 0.3$

(d) $\beta = 0.2, \delta = 0.01$ (e) $\beta = 0.2, \delta = 0.01$ (f) $\beta = 0.2, \delta = 0.01$

(g) $\beta = 0.1,\ \delta = 0.01$ (h) $\beta = 0.1,\ \delta = 0.01$ (i) $\beta = 0.1,\ \delta = 0.01$

Figure 2: Examples of adjacency matrices corresponding to the three structures and different levels of connectivity and reorganised by cluster.

The first community includes the first 100 nodes, while the second community comprises the remaining individuals. The second view is also made up of two communities, but this time the first community includes individuals from 1 to 50 and from 101 to 150 while the remaining individuals form the second community. The two views are visualized in Figure 3. In this case, the finest possible partition for this multidimensional network consists of four blocks (the first 50, the next 50, the following 50 and the last 50 individuals). However, when examining each view separately, we can only recover two blocks. This scenario highlights the necessity of integrating information from both views to achieve a more detailed and accurate clustering.



(a) First view

(b) Second view

Figure 3: Two adjacency matrices corresponding to scenario 1. Note that the two matrices can not be plotted in block by block at the same time since any permutation of the first one also affects the second one.

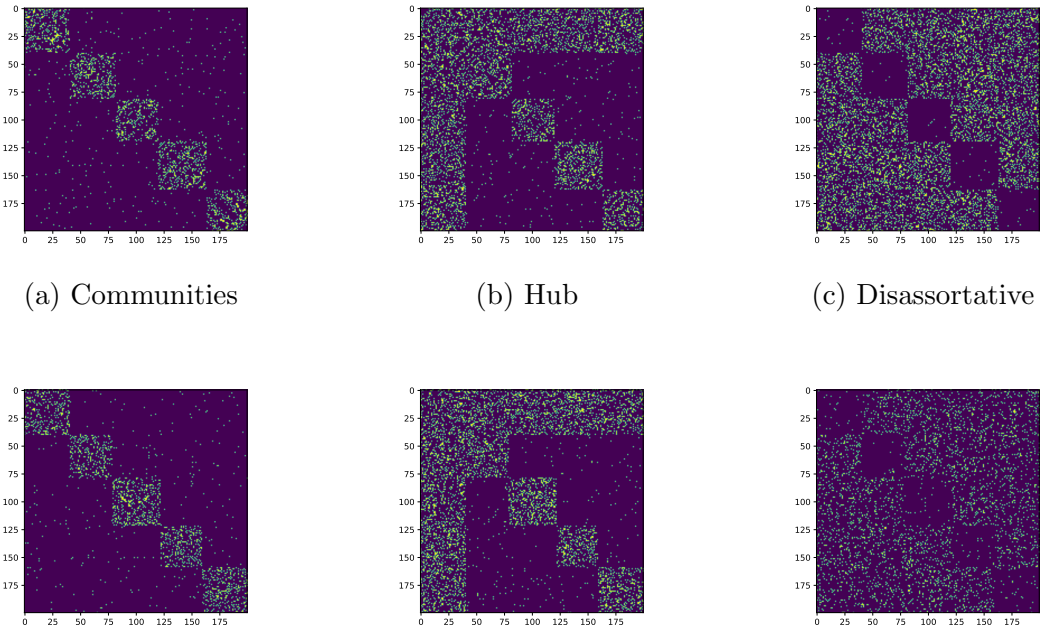
- **Scenario 2**: in this second scenario, we extended the analysis by considering four distinct network views. The first three correspond to the three structures discussed in Section 4.1: community, disassortative, and hub. The fourth view corresponds to an Erdős–Rényi random graph (Erdős and Renyi, 1959), which holds no clustering information, resulting in a 4-dimensional multi-graph. This scenario is designed to evaluate the effectiveness of our methods in capturing and distinguishing heterogeneous structures across multiple network views.

- **Scenario 3**: in this scenario, we considered four views. The first three views share the same underlying social structure, which can be modeled as a community, disassortative, or hub structure. Each view $A^{\ell)}$ is generated using parameter $\beta^{(l)}$ as the high probability connection and $\delta = 0.1$ as the low probability connection. This setup produces networks that, while preserving the same underlying structural pattern, may differ in their distributions. Such variability allows for a more dynamic analysis of evolving social relationships. Additionally, we include an Erdős–Rényi random graph (Erdős and Renyi, 1959), as in the previous scenario, to introduce a non-informative network view. In this case, the three informative networks convey identical information about the node partitions, demonstrating that Deep-LPBMM can effectively handle redundant information across multiple views.

Evaluation In Section 4.2, we evaluate Deep-LPBMM as a method for node clustering in these three structures. Recall that the generative model of Deep-LPBMM is designed to estimate partial memberships of nodes, rather than the "hard" cluster assignments typical of SBM. For clarity, we assign each node to the cluster whose estimated partial membership is the highest. A higher β indicates a more structured network, making it easier to identify the actual node cluster membership. It is important to note that the networks are generated following the SBM and our method is not given an advantage over competitors. The adjusted rand index (ARI, Yeung and Ruzzo, 2001) serves as our primary measure of clustering accuracy, reflecting the similarity between the true and inferred node partitions. An ARI of 0 suggests clustering is no better than random, while an ARI close to 1 indicates alignment with true node labels up to label switching.
We begin by presenting an illustrative example of Deep-LPBMM applied to a simulated multidimensional network.

4.2 Introductory example

Here, we focus on the case where we observe a multidimensional network with $L = 3$ views. For the shake of clarity, the three observed networks structures are: (i) a community-based view, (ii) a disassortative structure, and (iii) a hub configuration with connectivity levels $\beta = 0.2$ and $\delta = 0.01$.



(a) Communities (b) Hub (c) Disassortative

(d) Communities (e) Hub (f) Disassortative

Figure 4: The plots above are the three true adjacency matrix $A^{(1:3)}$ (respectively community, hub and disassortative) with the true connectivity matrix $\Pi^{(1:3)}$ and the plots below are the adjacency matrix $\hat{A}^{(1:3)}$ estimated by Deep-LPBMM. Note that there is a label switching between clusters in the second rows

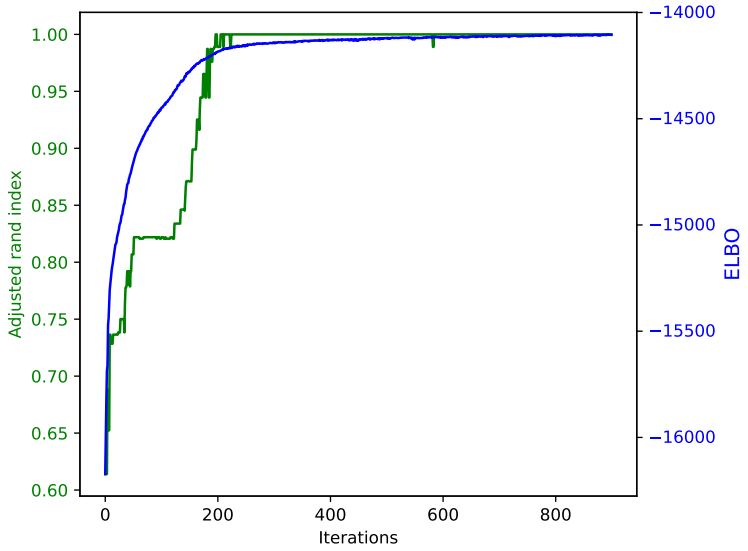


Figure 5: Evolution of the ARI and the ELBO during training.

The first metrics we consider are the evidence lower bound (ELBO) and the adjusted rand index (ARI). Even thought the ELBO does not allow to measure the performance of our methodology, it provides insights into the convergence of the optimisation routine, and the ARI measures the accuracy of the clustering results by comparing them to ground truth groupings. Figure 5 illustrates the evolution of the ELBO and ARI during the model's training process. Notably, both quantities clearly increase, indicating that as the optimization progresses, the model not only converges effectively but also improves its ability to correctly cluster nodes.

In Figure 4, we compare the true adjacency matrices of the simulated networks with those estimated by the Deep-LPBMM. Deep-LPBMM estimates the adjacency matrix $\hat{A}^{(\ell)}$ via the following distribution: $\hat{A}_{ij}^{(\ell)} \overset{\text{i.i.d.}}{\sim} \text{Bernoulli}(\hat{\eta}_i^\intercal \hat{\Pi}^{(\ell)} \hat{\eta}_j)$, where $\hat{\eta}$, and $\hat{\Pi}$ correspond to the estimates of Deep-LPBMM.
The visual similarity between these matrices is clear and the Frobenius norm of the differences between the true and estimated matrices is less than 10^{-2} underscoring the precision of our method in estimating the adjacency matrix $A^{(\ell)}$ and consequently the connectivity probabilities $\Pi^{(\ell)}$ and the latent block matrices Z. This small error demonstrates that Deep-LPBMM accurately recovers the structure of each layer of the network.

Additionally, in terms of interpretability, our method effectively captures the distinct structures of the three views. This ability to recover these specific configurations highlights the robustness of Deep-LPBMM in handling various structural patterns within complex, multidimensional networks.

Benchmarking

In this section, we compare the clustering performance of Deep-LPBMM against the stochastic block model for multiplex networks (Barbillon et al., 2017) in the three scenarios described in Section 4.1 with β varying between 0.1, 0.2, and 0.3. We recall that the higher the beta is, the more structured the network is and the easier it is to recover the true node cluster memberships. Additionally, to highlight the benefits of multidimensional network analysis, we include comparisons with models designed for single-layer networks: the stochastic block model

(SBM) and the deep latent position block model (Deep-LPBM). For these single-layer models, we randomly select one network layer for inference. All methodologies are provided with the true number of clusters, and the results are reported in Tables 2, 3 and 1 referring to respectively scenarios 1, 2 and 3. The best results are colored in blue and the second ones in green. When two results are equal up to the standard deviation, they are identically coloured.

$\beta^1 = 0.1$, $\beta^2 = 0.1$, $\beta^3 = 0.1$	ARI		
	Communities	Disassortative	Hub
Deep LPBM uni-view	0.29 ± 0.19	0.02 ± 0.02	0.10 ± 0.081
SBM uni-view	0.44 ± 0.31	0.06 ± 0.09	0.58 ± 0.24
SBM multi-view	0.94 ± 0.17	0.21 ± 0.361	0.85 ± 0.18
Deep-LPBMM	0.99 ± 0.01	0.112 ± 0.09	0.94 ± 0.07
$\beta^1 = 0.2$, $\beta^2 = 0.2$, $\beta^3 = 0.2$	ARI		
	Communities	Disassortative	Hub
Deep LPBM uni-view	0.72 ± 0.42	0.29 ± 0.21	0.71± 0.41
SBM uni-view	0.80 ± 0.38	0.79 ± 0.35	0.80 ± 0.36
SBM multi-view	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
Deep-LPBMM	1 ± 0.0	0.99 ± 0.001	1.00 ± 0.00
$\beta^1 = 0.3$, $\beta^2 = 0.2$, $\beta^3 = 0.1$	ARI		
	Communities	Disassortative	Hub
Deep-LPBM uniview	0.69 ± 0.45	0.52 ± 0.41	0.63 ± 0.45
SBM uni-view	0.70 ± 0.36	0.71 ± 0.21	0.66 ± 0.30
SBM multi-view	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0
Deep-LPBMM	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.05
$\beta^1 = 0.2$, $\beta^2 = 0.2$, $\beta^3 = 0.1$	ARI		
	Communities	Disassortative	Hub
Deep-LPBM uni-view	0.67 ± 0.37	0.32 ± 0.28	0.82±0.34
SBM uni-view	0.57 ± 0.44	0.51 ± 0.49	0.63 ± 0.38
SBM multi-view	1.00 ± 0.00	1.00 ± 0.00	0.95 ± 0.03
Deep-LPBMM	1 ± 0.001	0.99 ± 0.02	1.00 ± 0.00
$\beta^1 = 0.3$, $\beta^2 = 0.1$, $\beta^3 = 0.1$	ARI		
	Communities	Disassortative	Hub
Deep-LPBM uni-view	0.49 ± 0.33	0.31 ± 0.27	0.46 ± 0.4
SBM uni-view	0.57 ± 0.39	0.25 ± 0.41	0.52 ± 0.42
SBM multi-view	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
Deep-LPBMM	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00

Table 1: Benchmark to compare LPBM with competitors with three different connectivity levels on scenario 3.

$\beta = 0.2$ Deep-LPBM	0.35 ± 0.1
SBM uni-view	0.4 ± 0.05
SBM multi-view	1.00± 0.00
Deep-LPBMM	1.00 ± 0.0
$\beta = 0.3$ Deep-LPBM	0.39 ± 0.09
SBM uni-view	0.42 ± 0.1
SBM multi-view	1.00 ± 0.00
Deep-LPBMM	1.00 ± 0.00

Table 2: Benchmark to compare Deep-LPBMM with competitors on Scenario 1.

Scenario 1 For both difficulty levels, Deep-LPBMM perfectly recovers the true node partition, as does MSBM. In contrast, SBM and Deep-LPBM struggle significantly, recovering only two clusters instead of the required four. This highlights the limitations of single-layer models in capturing the complexity of multidimensional networks.

Scenario 2 for both $\beta = 0.3$ and $\beta = 0.2$, Deep-LPBMM and MSBM consistently achieve perfect clustering (ARI = 1) across all three structures allowing to retrieve all the different structures. Conversely, SBM and Deep-LPBM struggle, with less good performances. This is primarily because these models may select a non-informative network (e.g., the Erdős–Rényi network), which hinders clustering. However, the presence of this non informative network does not affect the performance of the Deep-LPBMM and MSBM.

When $\beta = 0.1$, all methods face challenges in recovering clusters. However, Deep-LPBMM and MSBM give a better partition than SBM, and Deep-LPBM in this difficult setting.

Scenario 3 When $\beta^1 = 0.3$ and independently to the values of β^2 and β^3, both Deep-LPBMM and MSBM achieve perfect clustering (ARI = 1) in all the community, disassortative and hub structures. SBM and Deep-LPBM and still have performances comparable to Scenario 2 due to the presence of non-informative view.

For more challenging cases where $\beta^1 = 0.2$, Deep-LPBM continues to cluster nodes effectively with an ARI of 1 in both community and hub structures. However, in the disassortative case, Deep-LPBMM performs slightly less than MSBM in retrieving clusters even if the gap of performance is very low.

Under very noisy conditions ($\beta^1 = 0.1, \beta^2 = 0.1$ and $\beta^3 = 0.1$), Deep-LPBMM outperforms all other methods in the community structure and performs comparably to MSBM in the hub configuration. However, none of the methods perform well in the disassortative structure,

	ARI
$\beta = 0.1$ Deep-LPBM uni-view	0.14 ± 0.16
SBM uni-view	0.19 ± 0.206
SBM multi-view	0.34 ± 0.38
Deep-LPBMM	0.35± 0.14
$\beta = 0.2$ Deep-LPBM uni-view	0.64 ± 0.36
SBM uni-view	0.78± 0.42
SBM multi-view	1.00± 0.00
Deep-LPBMM	1.00 ± 0.0
$\beta = 0.3$ Deep-LPBM uniview	0.87 ± 0.29
SBM uni-view	0.79 ± 0.43
SBM multi-view	1.00 ± 0.00
Deep-LPBMM	1.00 ± 0.00

Table 3: Benchmark to compare LPBM with competitors on Scenario 2.

highlighting the difficulty of this task. Node latent position-based methods often struggle with disassortative structures, especially when connectivity levels are low.

Our experiments demonstrate that Deep-LPBMM performs comparably to MSBM across all scenarios and even slightly outperforms it in certain cases. It is worth emphasizing again that these networks were generated using the SBM, which inherently favors MSBM over our method. In addition of this, Deep-LPBMM showcases significant strengths: it offers a direct visualization of clusters in a latent space, which MSBM lacks, and accommodates partial membership, allowing individuals to belong to multiple groups simultaneously. To conclude the analysis of these results we notice that Deep-LPBMM can also retrieve disassortative and hub patterns, configurations where position models usually struggle.

4.3 Model selection

This section provides an empirical analysis of the effectiveness of the Akaike Information Criterion (AIC) in determining the optimal number of clusters Q, within the Deep-LPBMM framework. Given the importance of accurately identifying the correct number of clusters, this evaluation aims to highlight the strengths of AIC as an objective metric in guiding model selection. We conducted our evaluation by applying Deep-LPBMM to the three previously introduced scenarios where all connectivity levels β are fixed to 0.12 and δ to 0.01. For each scenario, we generated 10 networks, true number of clusters in each network being $Q = 4$. To test the robustness of AIC in guiding model selection, we fitted the Deep-LPBMM model with several candidate values of Q: specifically, $Q = 2, 3, 4, 5, 6, 7, 10$ or 16. For each configuration, we ran the Deep-LPBMM algorithm with 10 different random initializations to account for variability. From the 10 runs, we selected the result corresponding to the highest AIC value, ensuring that the most optimized solution is retained for analysis.
In the evaluation, it may happen that the model fits a solution where one or more clusters are effectively unused, meaning that no nodes are assigned to those clusters. If such a scenario occurs, the model is reinterpreted as a model with $Q - 1$ clusters.
The results of this analysis are summarized in Table 4, where the performance of Deep-LPBMM across all tested scenarios and candidate values for Q is detailed. The results indicate that Deep-LPBMM consistently recovers the true number of clusters (i.e., 4 clusters) in many of the cases when guided by AIC.
We also note that when the problem is hard (disassortative configuration or Scenario 1), our model only finds the good number of clusters in more or less 50% of the cases but this was expected since the model struggles a lot in term of clustering performance in this configuration.
In conclusion, this evaluation highlights that AIC is a valuable tool for ensuring accurate model specification. These findings reinforce the practical utility of Deep-LPBMM in real-world applications, where the ability to recover nuanced cluster structures is essential for meaningful interpretation of complex multidimensional networks.

Q	Scen 1	Scen 2	Scen 3, communities	Scen 3, disassortative
1	0	0	0	0
2	1	0	0	0
3	0	0	0	2
4*	**7**	**8**	**10**	**5**
5	2	1	0	3
6	0	0	0	0
7	0	1	0	0
10	0	0	0	0
16	0	0	0	0

Table 4: Selection of the number of clusters by AIC. For each scenario, we simulated 10 networks with $Q = 4$ clusters.

5 Analysis of the Lazega lawyer dataset

5.1 The data

The Lazega lawyers dataset (Lazega, 2001) provides a rich and intricate representation of social dynamics within a professional legal environment. Collected between 1988 and 1991 at the "SG&R" corporate law firm in New England, this dataset captures the interactions and relationships of 71 lawyers working across three distinct network layers: advice exchange, friendship, and co-working ties. Each network layer offers a perspective on the professional and social fabric of the firm. The advice network, with a global density of 0.18, reflects professional consultations and guidance-seeking behavior, while the friendship network, with a lower density of 0.12, represents personal relationships outside of work. The co-working network, with a density of 0.15, illustrates collaboration on legal cases and professional tasks. Together, these networks provide a comprehensive view of both formal and informal interactions within the firm.

The dataset is more than just a snapshot of network ties: it encompasses several lawyer attributes, such as seniority, formal status (partners vs. associates), gender, office location, years with the firm, age, area of practice (litigation vs. corporate), and law school background. These attributes offer valuable context for analyzing the dynamics within the firm, especially the role of hierarchy, office location, and professional background in shaping network interactions.

In this section, we apply Deep-LPBMM to the Lazega lawyers dataset. Deep-LPBMM is well-suited for analyzing such multidimensional networks, as it leverages latent positions and block structures to capture partial membership and nuanced connectivity across the different network layers. By applying Deep-LPBMM to this dataset, we aim to uncover the latent social structure and partial group memberships that influence collaboration, advice-seeking, and social bonding within the firm. Through this analysis, we can better understand the interplay between professional and personal networks in shaping the lawyers' behaviors and interactions, offering a deeper insight into the mechanisms of cooperation and competition in a highly structured, high-pressure professional environment.

5.2 Deep-LPBMM analysis: model selection and visualisation

We begin this analysis by estimating the number of clusters using AIC. The results are presented in Figure 6 for Q varying from 2 to 15. It indicates that the highest AIC value is reached for Q equals to 5. Consequently, the rest of this section will present the corresponding results.
The dimension of the latent space is therefore equal to $Q-1=4$. To visualize this latent space on a 2-dimensional euclidean space, we use the TSNE projection algorithm (Arora, Hu, and Kothari, 2018).
The networks in Figure 7 are obtained by projecting η (partial membership probabilities) estimated by Deep-LPBMM. Connections between the lawyers in the different layers are represented as grey lines joining pairs of latent coordinates and the node pie-charts, in the Figure 7 on the right-hand side, represent the estimated partial memberships. In addition the estimated $(\hat{\Pi}^{(\ell)})^l$ matrix are displayed in Figure 8 and in Figure 9, we plotted box-plots and bar-plots to show how the numerical attributes of lawyers are split into the different clusters

5.3 Analysis of the clusters

The nodes in cluster 1 tend to exchange advice within the cluster more frequently (with a probability of about 0.5) and maintain more friendships (about 0.4 probability) between them than with nodes from other clusters. This is an indication of a community structure. They also have friendship bounds and advice exchanging with Cluster 2. In terms of co-working third view, nodes in Cluster 1 are more likely to collaborate with nodes in other clusters, especially with individuals in cluster 2, suggesting a disassortative pattern. Moreover, individuals in this group share common attributes: they are all associates, work in the Boston office, are relatively new to the firm (average tenure of two years) and are younger (average age of 33 years). More importantly, this cluster would be hard to find without taking all the three networks into account. Indeed, members in Cluster 1 react similarly to Cluster 2 in term of advice exchanging and friendship. To differentiate them, it is necessary to consider the third view where these two clusters are well separated. Indeed, in term of co-working, nodes in cluster in Cluster 1 barely interact between themselves and nodes in Cluster 2 frequently work together. The nodes in cluster 2 exchange advice primarily among themselves and have a particularly high probability of friendship within the group (about 0.7). Across all three views, Cluster 2 exhibits a clear community structure. In addition, all members are based in the Boston office and are litigation lawyers. This group has the highest average age (45).
Nodes in Cluster 3 form a strong community across all the three dimension. they all have their office located in Hatford. Some members of this cluster have partial membership to Cluster 5 which is fully composed of corporate lawyers. We notice that individuals in Cluster 3 with high partial belongship to Cluster 5 are also corporate lawyers.
Cluster 4 consists of individuals with a somewhat diverse composition. As shown in Figure 7, in this cluster, several nodes partially belong to many clusters. Specially corporate lawyers in this cluster partially belong to Cluster 5 and lawyers based at Hartford in this cluster partially belong to Cluster 3. More importantly it seems difficult to retrieve this cluster by only considering the co-working ties since these individuals almost do not cowork with others. It is important to note that MSBM does not allow to directly have the latent visualisation available in Figure 7 and can not deal with partial membership that often occur in this dataset.

Moreover, if we only looked at one of the three networks, it would be almost impossible to find these five clusters, as groups 1 and 2 would collapse or cluster 4 would disappear.

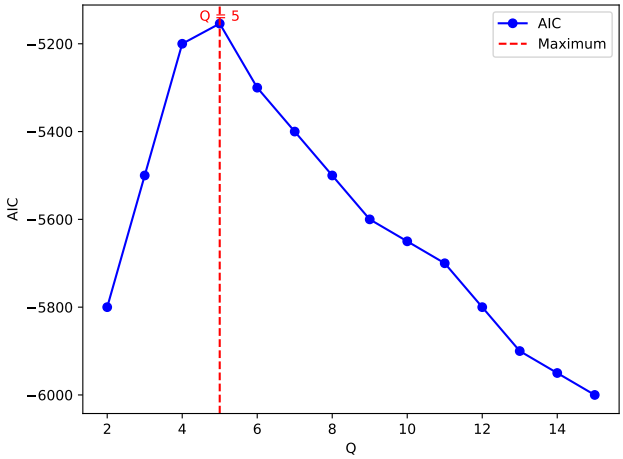
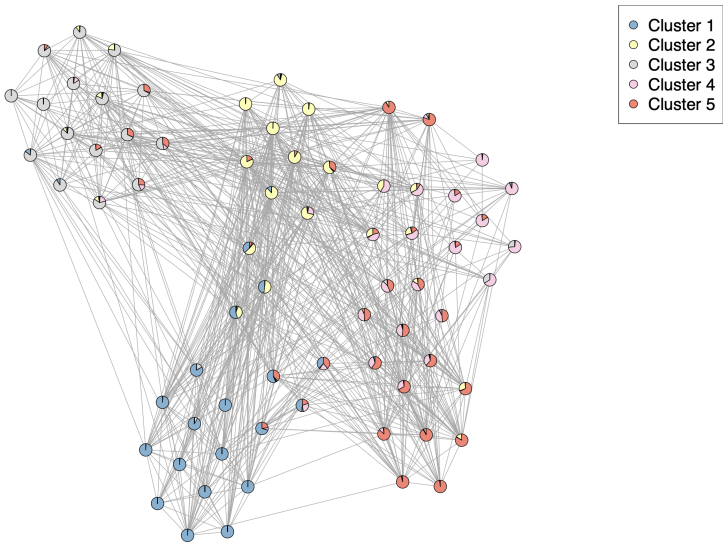
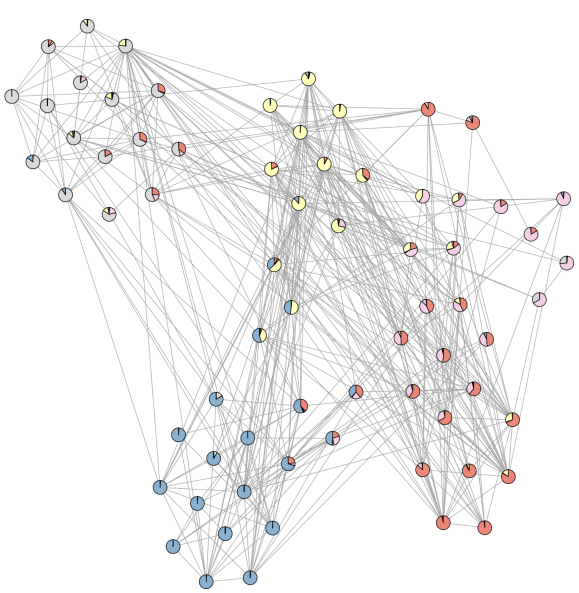


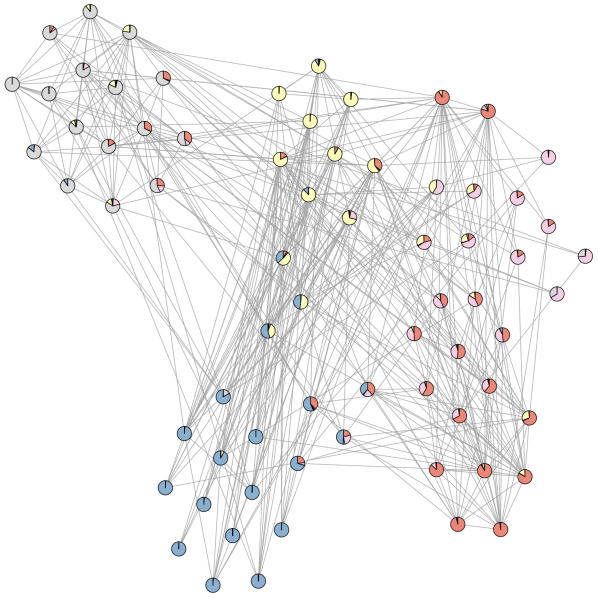
Figure 6: Evolution of AIC values of Deep-LPBM for Q varying from 2 to 15 in the Lazega lawyer dataset.



(a) Advice network

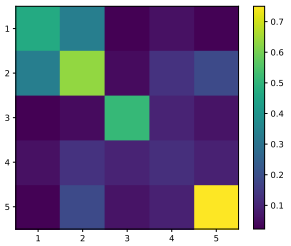


(b) Friendship network

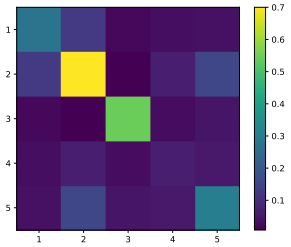


(c) work network

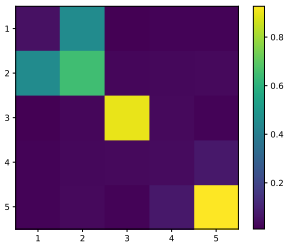
Figure 7: Clustering visualization of the multilayer network by Deep-LPBMM with partial membership effects.



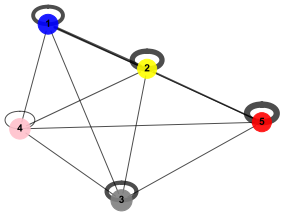
(a) $\Pi^{(1)}$



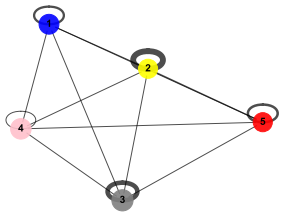
(b) $\Pi^{(2)}$



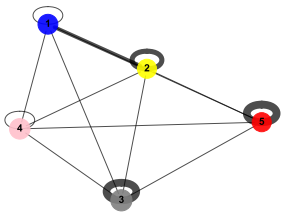
(c) $\Pi^{(3)}$



(d) $\Pi^{(1)}$

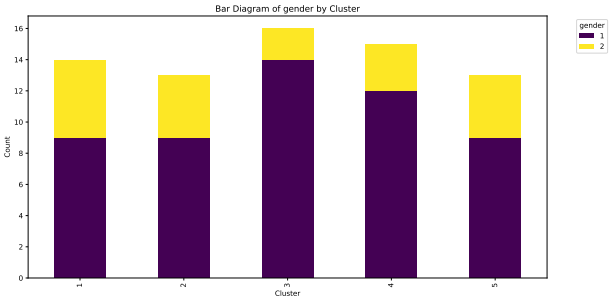


(e) $\Pi^{(2)}$

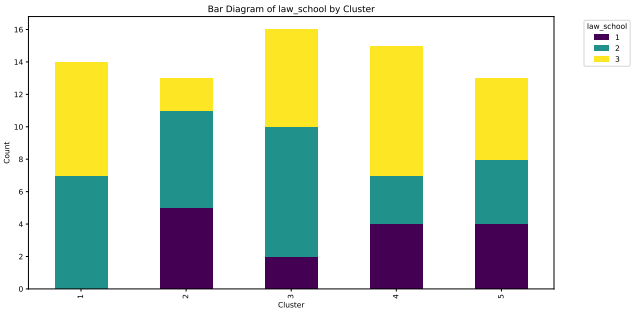


(f) $\Pi^{(3)}$

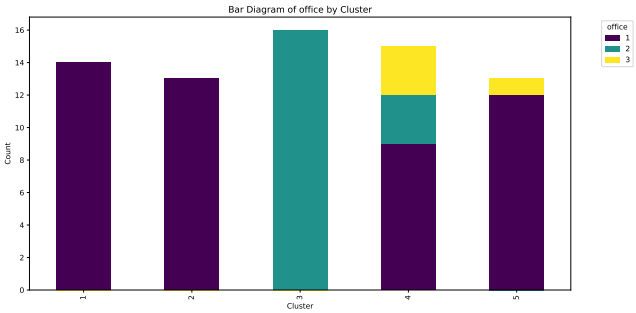
Figure 8: Visualization of the $\Pi^{(\ell)}$ matrices estimated by the Deep-LPBMM (first row). In the three figures i the second row (d, e, f) nodes represent the different clusters. The size of the nodes depends on the size of the clusters. The size of the edges is proportional to the linkage probability.



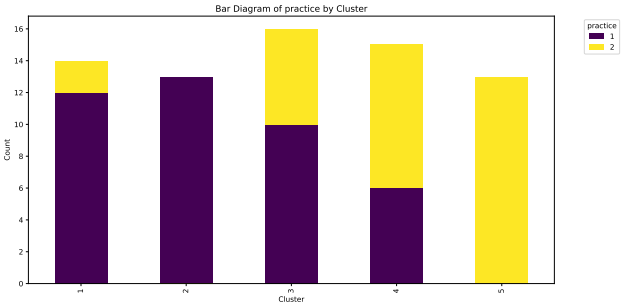
(a) Gender: Man vs Woman



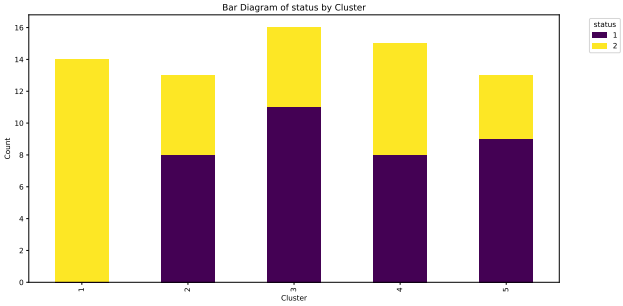
(b) Law school: Harvard, Yale vs Ucon vs Other



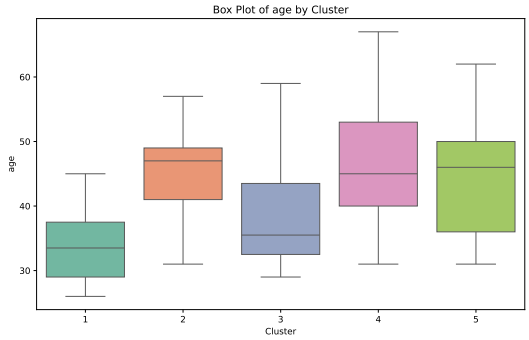
(c) Office: Boston vs Providence vs Hartford



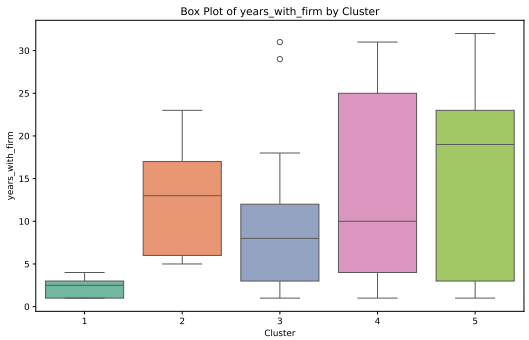
(d) Practice : Litigation vs Corporate



(e) Status: Associate vs Partner



(f) Age



(g) Year with the firm

Figure 9: Visualization of the distribution of the attributes cluster by cluster.

6 Conclusion

This paper extended the work of Boutin, Latouche, and Bouveyron (2024), a new methodology combining a block model with a deep latent position model adapted to multidimensional networks. By modifying the edge distribution and marginalizing over a bijective transformation of the node latent representations and introducing view-depending block connections, we managed to use the same node embedding for all the views and considered them as cluster probability memberships. We obtained richer results, providing a full network representation, to incorporate details at the node-level. Deep-LPBMM is based on the encoder of a graph variational auto-encoder and multi layer perceptions combined with block models decoder. Experiments showed that on communities, hubs and disassortative networks, our methodology rightfully translated the network salient information into the latent space. An extensive study with simulations illustrated the capacity of our methodology to retrieve clusters.

Acknowledgments

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.

References

Airoldi, Edoardo M et al. (2006). "Mixed membership stochastic block models for relational data with application to protein-protein interactions". In: *Proceedings of the international biometrics society annual meeting*. Vol. 15, p. 1.

Akaike, Hirotugu (1974). "A new look at the statistical model identification". In: *IEEE transactions on automatic control* 19.6, pp. 716–723.

Arora, S, W Hu, and P K Kothari (2018). "An analysis of the t-sne algorithm for data visualization". In: *Conference on learning theory*. PMLR, pp. 1455–1462.

Barbillon, Pierre et al. (2017). "Stochastic block models for multiplex networks: an application to a multilevel network of researchers". In: *Journal of the Royal Statistical Society Series A: Statistics in Society* 180.1, pp. 295–314.

Biernacki, Christophe, Gilles Celeux, and Gérard Govaert (2000). "Assessing a mixture model for clustering with the integrated completed likelihood". In: *IEEE transactions on pattern analysis and machine intelligence* 22.7, pp. 719–725.

Boutin, Rémi, Pierre Latouche, and Charles Bouveyron (2023). "The Deep Latent Position Topic Model for Clustering and Representation of Networks with Textual Edges". In: *arXiv preprint arXiv:2304.08242*.

— (2024). "The Deep Latent Position Block Model For The Block Clustering And Latent Representation Of Networks". In: *arXiv preprint arXiv:2412.01302*.

Bouveyron, Charles et al. (2019). *Model-based clustering and classification for data science: with applications in R*. Vol. 50. Cambridge University Press.

De Santiago, Kylliann, Marie Szafranski, and Christophe Ambroise (2024). "Mixture of multilayer stochastic block models for multiview clustering". In: *arXiv preprint arXiv:2401.04682*.

D'Angelo, Silvia, Marco Alfò, and Michael Fop (2023). "Model-based clustering for multidimensional social networks". In: *Journal of the Royal Statistical Society Series A: Statistics in Society* 186.3, pp. 481–507.

D'Angelo, Silvia, Thomas Brendan Murphy, and Marco Alfò (2019). "Latent space modelling of multidimensional networks with application to the exchange of votes in Eurovision song contest". In.

Erdős, OSP and A Renyi (1959). "On random graphs". In: *Publ. Math* 6, pp. 290–297.

Fred, Ana LN and Anil K Jain (2005). "Combining multiple clusterings using evidence accumulation". In: *IEEE transactions on pattern analysis and machine intelligence* 27.6, pp. 835–850.

Gollini, Isabella and Thomas Brendan Murphy (2016). "Joint modeling of multiple network views". In: *Journal of Computational and Graphical Statistics* 25.1, pp. 246–265.

Han, Qiuyi, Kevin Xu, and Edoardo Airoldi (2015). "Consistent estimation of dynamic and multi-layer block models". In: *International Conference on Machine Learning*. PMLR, pp. 1511–1520.

Handcock, Mark S, Adrian E Raftery, and Jeremy M Tantrum (2007). "Model-based clustering for social networks". In: *Journal of the Royal Statistical Society Series A: Statistics in Society* 170.2, pp. 301–354.

Heller, Katherine A, Sinead Williamson, and Zoubin Ghahramani (2008). "Statistical models for partial membership". In: *Proceedings of the 25th International Conference on Machine learning*, pp. 392–399.

Hoff, Peter D, Adrian E Raftery, and Mark S Handcock (2002). "Latent space approaches to social network analysis". In: *Journal of the american Statistical association* 97.460, pp. 1090–1098.

Holland, Paul W, Kathryn Blackmond Laskey, and Samuel Leinhardt (1983). "Stochastic blockmodels: First steps". In: *Social networks* 5.2, pp. 109–137.

Holland, Paul W and Samuel Leinhardt (1981). "An exponential family of probability distributions for directed graphs". In: *Journal of the american Statistical association* 76.373, pp. 33–50.

Kingma, Diederik P, Max Welling, et al. (2019). "An introduction to variational autoencoders". In: *Foundations and Trends® in Machine Learning* 12.4, pp. 307–392.

Kipf, T N and M Welling (2016a). "Semi-supervised classification with graph convolutional networks". In: *arXiv preprint arXiv:1609.02907*.

Kipf, Thomas N and Max Welling (2016b). "Variational graph auto-encoders". In: *arXiv preprint arXiv:1611.07308*.

Latouche, Pierre, Etienne Birmelé, and Christophe Ambroise (2009). "Overlapping stochastic block models". In: *arXiv preprint arXiv:0910.2098*.

Lazega, Emmanuel (2001). *The collegial phenomenon: The social mechanisms of cooperation among peers in a corporate law partnership*. Oxford University Press, USA.

Liang, Dingge et al. (2022). "Deep latent position model for node clustering in graphs". In: *ESANN 2022-30th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*.

Mehta, Nikhil, Lawrence Carin Duke, and Piyush Rai (2019). "Stochastic blockmodels meet graph neural networks". In: *International Conference on Machine Learning*. PMLR, pp. 4466–4474.

Monti, Stefano et al. (2003). "Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data". In: *Machine learning* 52, pp. 91–118.

Nowicki, Krzysztof and Tom A B Snijders (2001). "Estimation and prediction for stochastic blockstructures". In: *Journal of the American statistical association* 96.455, pp. 1077–1087.

Pan, Shirui et al. (2018). "Adversarially regularized graph autoencoder for graph embedding". In: *arXiv preprint arXiv:1802.04407*.

Paul, Subhadeep and Yuguo Chen (2016). "Consistent community detection in multi-relational data through restricted multi-layer stochastic blockmodel". In.

Rebafka, Tabea (2024). "Model-based clustering of multiple networks with a hierarchical algorithm". In: *Statistics and Computing* 34.1, p. 32.

Rezende, Danilo and Shakir Mohamed (2015). "Variational inference with normalizing flows". In: *International conference on machine learning*. PMLR, pp. 1530–1538.

Robins, Garry et al. (2007). "Recent developments in exponential random graph (p*) models for social networks". In: *Social networks* 29.2, pp. 192–215.

Salter-Townshend, Michael and Thomas Brendan Murphy (2013). "Variational Bayesian inference for the latent position cluster model for network data". In: *Computational Statistics & Data Analysis* 57.1, pp. 661–671.

Schaeffer, Satu Elisa (2007). "Graph clustering". In: *Computer science review* 1.1, pp. 27–64.

Sewell, Daniel K and Yuguo Chen (2017). "Latent space approaches to community detection in dynamic networks". In.

Snijders, Tom AB and Krzysztof Nowicki (1997). "Estimation and prediction for stochastic blockmodels for graphs with latent block structure". In: *Journal of classification* 14.1, pp. 75–100.

Stanley, Natalie et al. (2016). "Clustering network layers with the strata multilayer stochastic block model". In: *IEEE transactions on network science and engineering* 3.2, pp. 95–105.

Wasserman, Stanley and Philippa Pattison (1996). "Logit models and logistic regressions for social networks: I. An introduction to Markov graphs and p". In: *Psychometrika* 61.3, pp. 401–425.

Yeung, Ka Yee and Walter L Ruzzo (2001). "Details of the adjusted rand index and clustering algorithms, supplement to the paper an empirical study on principal component analysis for clustering gene expression data". In: *Bioinformatics* 17.9, pp. 763–774.