The Deep Latent Position Block Model for Block Clustering and Latent Representation of Nodes in Networks

Investigation of Boutin et al. (2025) — Project Poster

Yannaël Bossard — Karina Musina

Introduction

The growing availability of network data in social sciences and biology calls for powerful tools for both analysis and visualisation. Existing approaches mainly fall into two families:

- **positional models**, which embed nodes in a Euclidean space but struggle to represent important real-world structures disassortative patterns
- **block models**, which capture rich connectivity patterns between groups but do not provide meaningful representation in an Euclidean space

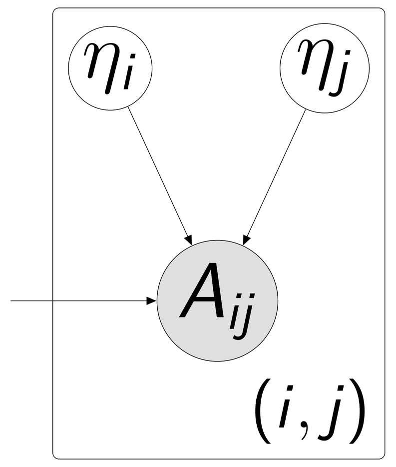
Deep LPBM, proposed by Boutin et al. [1], aims to bridge this gap by combining block modeling with latent positional representations in a unified probabilistic framework.

Main contributions

- A new structured decoder, Deep LPBM, capable of modelling **any type of connectivity** pattern.
- The model is able to capture **partial** memberships.
- The model handles heterogeneous **blocks** within a latent **positional** framework

Model assumptions

Nodes. The network is modelled by an adjacency matrix $A \in \{0,1\}^{N\times N}$ on N nodes.
Partial memberships. Nodes are grouped into Q clusters. We denote by Δ_Q the probability simplex. The partial membership vector η_i of node i belongs to Δ_Q.
Each node i is associated with a latent Gaussian vector $z_i \in \mathbb{R}^{Q-1}$, mapped to the simplex through $\eta_i = softmax(z_i)$



Graphical model of the Deep LPBM.

Connectivity by blocks. A symmetric block connectivity matrix $\Pi \in [0,1]^{Q\times Q}$ specifies the probability of connection between blocks. Conditionally on (η_i, η_j, Π), edges are assumed independent and

$$A_{ij} \mid \eta_i, \eta_j, \Pi \sim \mathrm{Bernoulli}\left(\eta_i^\top \Pi\, \eta_j\right)$$

Edges. We define $P = (P_{ij})_{i,j=1}^N$ as the $N\times N$ symmetric probability matrix such that

$$P_{ij} = \eta_i^\top \Pi\, \eta_j.$$

References

[1] Rémi Boutin, Pierre Latouche, and Charles Bouveyron. "The Deep Latent Position Block Model for Block Clustering and Latent Representation of Nodes in Networks". In: *Statistics and Computing* 35.5 (2025), p. 151.

Variational inference and optimisation

We aim to estimate the connectivity matrix Π by maximising the marginal likelihood

$$\log p(A \mid \Pi) = \log \int p(A, Z \mid \Pi) dZ.$$

This integral is **intractable** due to the softmax mapping and the fact that each A_{ij} depends on two latent variables (z_i, z_j), which makes the posterior globally coupled. We therefore use a **variational EM approach.**

We introduce a variational distribution $R(Z)$ as an approximation to $p(Z \mid A, \Pi)$, yielding

$$\log p(A \mid \Pi) \geq \mathcal{L}(\Pi; R),$$

where the ELBO is

$$\mathcal{L}(\Pi; R) = \mathbb{E}_{R(Z)}[\log p(A \mid Z, \Pi)] - \mathrm{KL}(R(Z)\|p(Z)).$$

We then restrict the variational family by assuming a mean-field factorisation and an encoder-based parametrisation:

$$R_\phi(Z) = \prod_{i=1}^{N} \mathcal{N}(z_i; \mu_{\phi,i}(A), \sigma^2_{\phi,i}(A) I_d),$$

where $(\mu_{\phi,i}(A), \sigma^2_{\phi,i}(A))$ are produced by a GCN encoder.

The KL term is computed in closed form, while the reconstruction expectation is approximated by Monte Carlo using the reparameterisation trick enabling stochastic gradient optimisation:

$$\varepsilon_i \sim \mathcal{N}(0, I_d), \qquad z_i = \mu_{\phi,i}(A) + \sigma_{\phi,i}(A) \odot \varepsilon_i,$$

Practical training

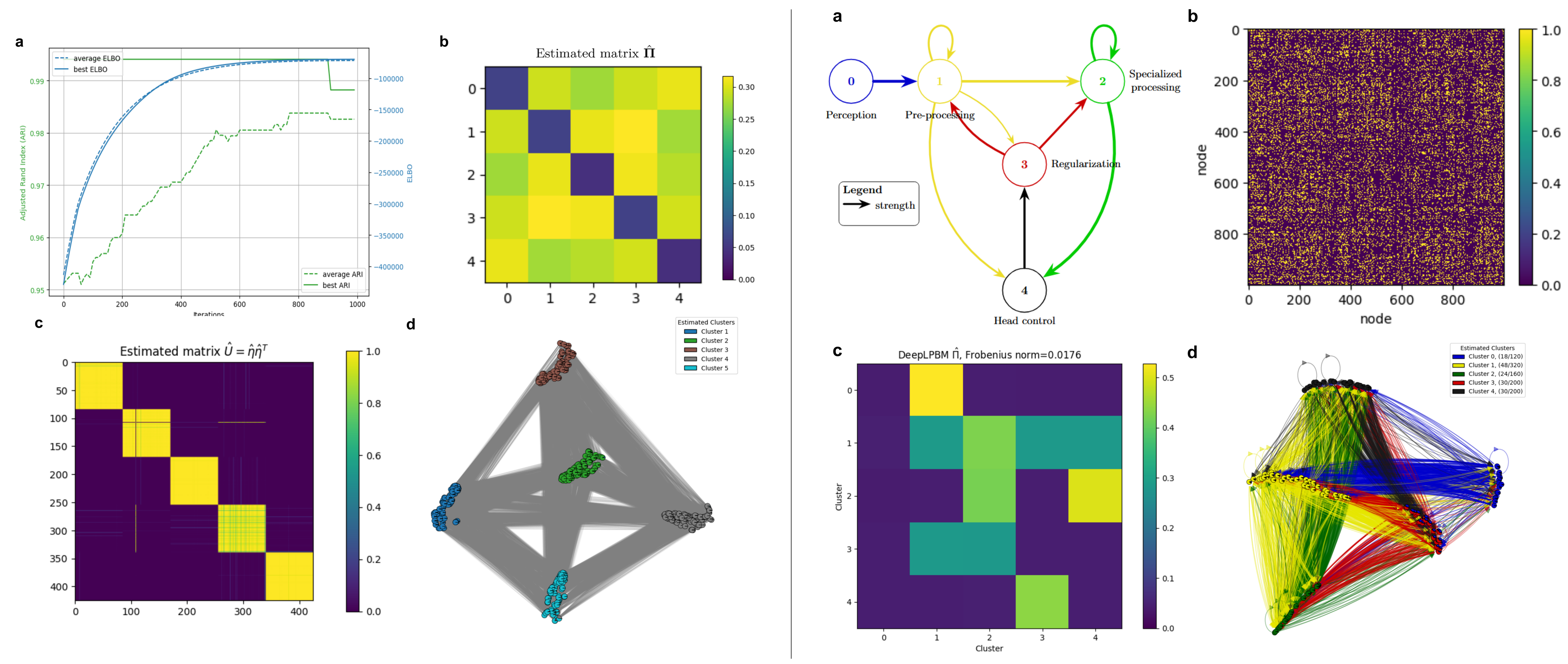
- **Encoder** GCN encoder maps the adjacency matrix A to variational parameters $(\mu_\phi(A), \log \sigma_\phi(A))$
- **Decoder** Given sampled soft memberships $\hat{\eta}_i = softmax(z_i)$ and connectivity parameters Π, the decoder reconstructs edge probabilities $\hat{P}_{ij} = \hat{\eta}_i^\top \Pi\, \hat{\eta}_j$ (and $A_{ij} \sim \mathrm{Bernoulli}(\hat{P}_{ij})$), which forms the data-fit term in the ELBO.
- **Initialization.** We use K-means on A to get hard labels c_i then invert the softmax to obtain z_i^0. We pretrain the encoder to match $\mu_\phi(A)_i \approx z_i^0$ with small variances, and initialise Π.
- **Optimisation.** We jointly optimise ϕ (encoder) and $\tilde{\Pi}$ (decoder) with Adam optimizer.
- **Model selection.** We train the model using several random seeds and keep the run with the highest ELBO. Empirically, AIC gave the most reliable results for selecting the number of clusters, so we adopt it:

$$\mathrm{AIC} = \log p(A|\hat{Z}, \hat{\Pi}) - \frac{Q(Q+1)}{2} - N(Q-1)$$

Extension to Directed Graph-Structured Data

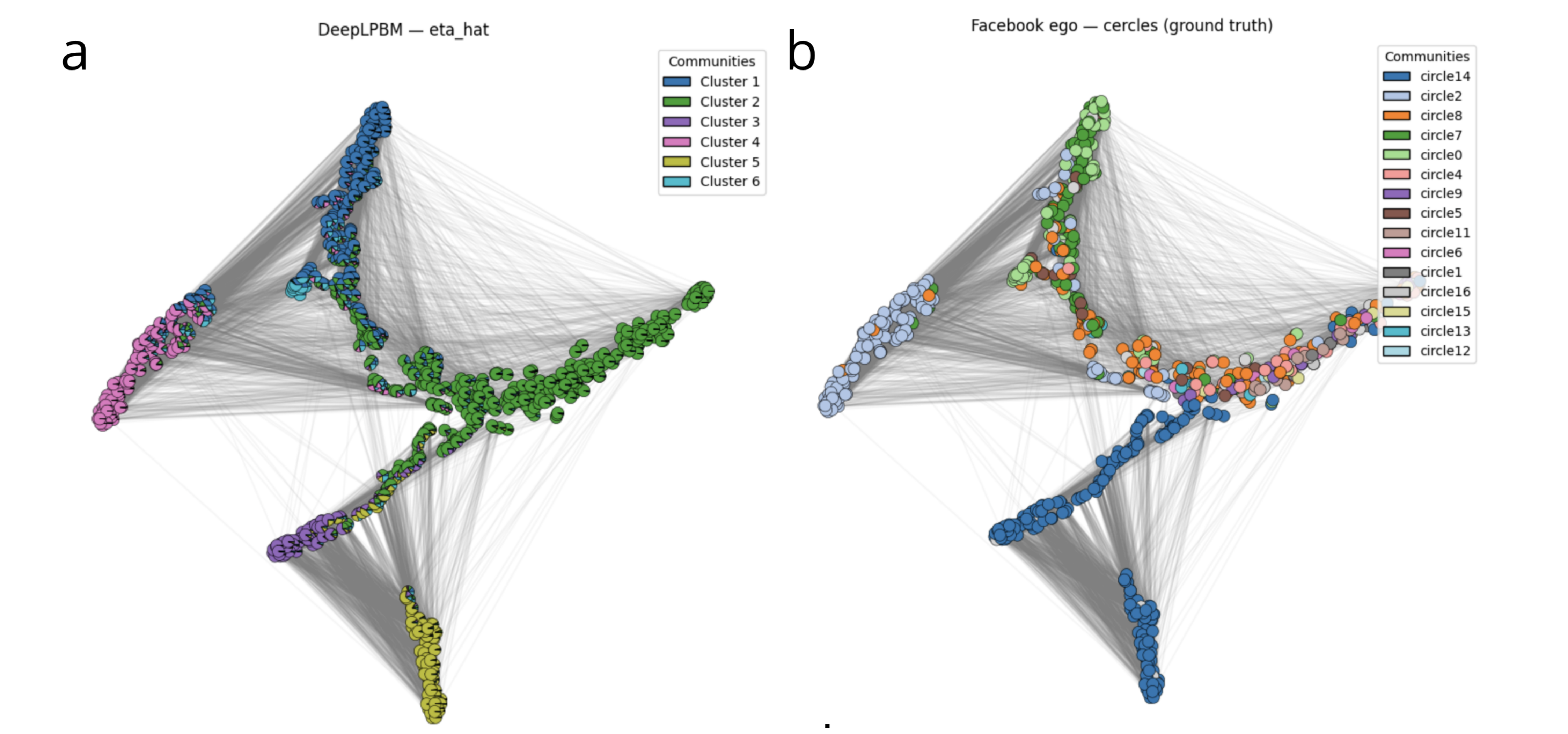
- Remove symmetric assumption $P(A_{ij} = 1|\eta_i, \eta_j, \Pi) = \eta_i^\top \Pi \eta_j \neq \eta_j^\top \Pi \eta_i$
- Use of GAT instead of GCN which is able to naturally handle directed edges

Results: Undirected & Directed synthetic graph



Frobenius norm error with Π^* ($\beta = 0.4$, $N = 350$)	Assortative	Disassortative	Hub
Π^0 (Init)	0.028 ± 0.004	0.032 ± 0.008	0.033 ± 0.004
$\hat{\Pi}$ (Est.)	0.34 ± 0.02	0.09 ± 0.02	0.32 ± 0.03

Real-world graph



Conclusion & Future Work

Deep LPBM appears to be a promising framework that bridges block models and latent position models. However, we were unable to reproduce the performance reported in the paper, and training proved to be unstable. This lack of reproducibility limits the method's applicability to real-world datasets.
Our extension to directed graphs is encouraging and suggests that the framework can generalize. Finally, the paper does not discuss how to leverage node-level information (when available), which could be an important avenue to improve performance on many practical datasets.