Received: 16 August 2023 | Revised: 13 May 2024 | Accepted: 1 September 2024

DOI: 10.1111/sjos.12754

ORIGINAL ARTICLE

Scandinavian Journal of Statistics

Model-based clustering in simple hypergraphs through a stochastic blockmodel

Luca Brusa[1] | **Catherine Matias**[2,3,4]

[1]Department of Statistics and Quantitative Methods, University of Milano-Bicocca, Milano, Italy

[2]Laboratoire de Probabilités, Statistique et Modélisation, Sorbonne Université, Paris, France

[3]Université de Paris Cité, Paris, France

[4]Centre National de la Recherche Scientifique, Paris, France

Correspondence
Luca Brusa, Department of Statistics and Quantitative Methods, University of Milano-Bicocca, Via Bicocca degli Arcimboldi 8, 20126 Milano, Italy.
Email: luca.brusa@unimib.it

Funding information
Ministero dell'Università e della Ricerca, Grant/Award Number: PRIN 2022TZEXKF; Agence Nationale de la Recherche (EcoNet), Grant/Award Number: ANR-18-CE02-0010-01

Abstract

We propose a model to address the overlooked problem of node clustering in simple hypergraphs. Simple hypergraphs are suitable when a node may not appear multiple times in the same hyperedge, such as in co-authorship datasets. Our model generalizes the stochastic blockmodel for graphs and assumes the existence of latent node groups and hyperedges are conditionally independent given these groups. We first establish the generic identifiability of the model parameters. We then develop a variational approximation Expectation-Maximization algorithm for parameter inference and node clustering, and derive a statistical criterion for model selection. To illustrate the performance of our `R` package `HyperSBM`, we compare it with other node clustering methods using synthetic data generated from the model, as well as from a line clustering experiment and a co-authorship dataset.

KEYWORDS

co-authorship network, high-order interactions, latent variable model, line clustering, non-uniform hypergraph, variational expectation-maximization

1 | INTRODUCTION

Over the past two decades, a wide range of models has been developed to capture pairwise interactions represented in graphs. However, modern applications in various fields have highlighted the necessity to consider high-order interactions, which involve groups of three or more nodes. Simple examples include triadic and larger group interactions in social networks (whose importance has been recognized early on, see Wolff, 1950), scientific co-authorship (Estrada & Rodríguez-Velázquez, 2006), interactions among more than two species in ecological systems (Muyinda et al., 2020; Singh & Baruah, 2021), or high-order correlations between neurons in brain networks (Chelaru et al., 2021). To formalize these high-order interactions, hypergraphs provide the most general framework. Similar to a graph, a hypergraph consists of a set of nodes and a set of hyperedges, where each hyperedge is a subset of nodes involved in an interaction. In this context, it is important to distinguish simple hypergraphs from *multiset hypergraphs*, where hyperedges can contain repeated nodes. Multisets are a generalization of sets, allowing elements to appear with varying multiplicities. Recent reviews on high-order interactions can be found in the works of Battiston et al. (2020), Bick et al. (2023), and Torres et al. (2021).

Despite the increasing interest in high-order interactions, the statistical literature on this topic remains limited. Some graph-based statistics, such as centrality or clustering coefficient, have been extended to hypergraphs to aid in understanding the structure and extracting information from the data (Estrada & Rodríguez-Velázquez, 2006). However, these statistics do not fulfill the need for random hypergraph models. Early analyses of hypergraphs have relied on their embedding into the space of bipartite graphs (see, e.g., Battiston et al., 2020). Hypergraphs with self-loops and multiple hyperedges (weighted hyperedges with integer-valued weights) are equivalent to bipartite graphs. However, bipartite graph models were not specifically designed for hypergraphs and may introduce artifacts; we refer to Section A in the Appendix S1 for more details.

Generalizing Erdős-Rényi's model of random graphs leads to uniformly random hypergraphs. This model involves drawing uniformly at random from the set of all m-uniform hypergraphs (hypergraphs with hyperedges of fixed cardinality m) over a set of n nodes. However, similar to Erdős–Rényi's model for graphs, this hypergraph model is too simplistic and homogeneous to be used for statistical analysis of real-world datasets. In the configuration model for random graphs, the graphs are generated by drawing uniformly at random from the set of all possible graphs over a set of n nodes, while satisfying a given prescribed degree sequence. In the context of hypergraphs, configuration models were proposed in Ghoshal et al. (2009), focusing on tripartite and 3-uniform hypergraphs. Later, Chodrow (2020) extended the configuration model to a more general hypergraph setup. In these references, both the node degrees and the hyperedge sizes are kept fixed (a consequence of the fact that they rely on bipartite representations of hypergraphs). The configuration model is useful for sampling (hyper)graphs with the same degree sequence (and hyperedge sizes) as an observed dataset through shuffling algorithms. Therefore, it is often employed as a null model in statistical analyses. However, sampling exactly (rather than approximately) from this model poses challenges, particularly in the case of hypergraphs. For a comprehensive discussion on this issue, we refer readers to Sect. 4 in Chodrow (2020).

Another popular approach for extracting information from heterogeneous data is clustering. In the context of graphs, stochastic blockmodels (SBMs) were introduced in the early eighties (Frank & Harary, 1982; Holland et al., 1983) and have since evolved in various directions. These models assume that nodes are grouped into clusters, and the probabilities of connections between nodes are determined by their cluster memberships. Variants of SBMs have been developed to

handle weighted graphs and degree-corrected versions, among others. In the context of hypergraphs, Ghoshdastidar and Dukkipati (2017) studied a spectral clustering approach based on a hypergraph Laplacian, and obtained its weak consistency under a Hypergraph SBM (HSBM) with certain restrictions on the model parameters. More recently, Deng et al. (2024) established the strong consistency of the basic spectral clustering under the degree-corrected HSBM (DCHSBM) in the sparse regime where the maximum expected hyperdegree might be of order $\Omega(\log n)$ and n is the number of nodes. By introducing hypergraphons, Balasubramanian (2021) extended the ideas of hypergraph SBMs to a nonparametric setting. In a parallel vein, Turnbull et al. (2023) proposed a latent space model for hypergraphs, generalizing random geometric graphs to hypergraphs, although it was not specifically designed to capture clustering. An approach linked to SBMs is presented in Vazquez (2009), where nodes belong to latent groups and participate in a hyperedge with a probability that depends on both their group and the specific hyperedge.

Modularity is a widely used criterion for clustering entities in the context of interaction data. It aims to identify specific clusters, known as communities, characterized by high within-group connection probabilities and low between-group connection probabilities (Ghoshdastidar & Dukkipati, 2014). However, in the hypergraph context, the definition of modularity is not unique. In particular, Kamiński et al. (2019) introduced a "strict" modularity criterion, where only hyperedges with all their nodes belonging to the same group contribute to an increase in modularity. Their criterion measures the deviation of the number of these homogeneous hyperedges from a new null model called the configuration-like model for hypergraphs, where the average values of the degrees are fixed. Building upon this, Chodrow et al. (2021) proposed a degree-corrected hypergraph SBM and introduced two new modularity criteria. Similar to Kamiński et al. (2019), one of these criteria utilizes an "all-or-nothing" affinity function that distinguishes whether a given hyperedge is entirely contained within a single cluster or not. In this setup, they established a connection between approximate maximum likelihood estimation and their modularity criterion. This work is reminiscent of the work of Newman (2016) in the graph context. However, the estimators proposed by Chodrow et al. (2021) do not guarantee maximum likelihood estimation, as the parameter space is constrained by assuming a symmetric affinity function. We refer to Poda and Matias (2024) for an empirical comparison of these modularity-based methods.

It is important to highlight that the developments presented in Kamiński et al. (2019) and Chodrow et al. (2021) are specifically conducted in the context of multiset hypergraphs, where hyperedges can contain repeated nodes with certain multiplicities. The use of multiset hypergraphs simplifies some of the challenges associated with computing modularity. However, to the best of our knowledge, modularity approaches still lack instantiation in the case of simple hypergraphs where each node can only appear once in a hyperedge. More specifically, the null model used in hypergraph modularity criteria relies on a model for multiset hypergraphs, similar to how the null model used in classical graph modularity is based on graphs with self-loops. While it is known in the case of graphs that this assumption is inadequate, as it induces a stronger deviation than expected and affects sparse networks as well (Cafieri et al., 2010; Massen & Doye, 2005; Squartini & Garlaschelli, 2011), the assumption of multisets has not yet been discussed in the context of hypergraph modularity.

In the context of community detection, random walk approaches have also been utilized for hypergraph clustering (Swan & Zhan, 2021). Additionally, low-rank tensor decompositions have been explored (Ke et al., 2020). The misclassification rate for the community detection problem in hypergraphs and its limits have been analyzed in various contexts (see, for instance, Ahn et al., 2018; Chien et al., 2019; Cole & Zhu, 2020). It is worth mentioning that a recent approach

has been proposed to cluster hyperedges instead of nodes (Ng & Murphy, 2022). However, our focus in this work is on clustering nodes.

The literature on high-order interactions often discusses simplicial complexes alongside hypergraphs (Battiston et al., 2020). However, the unique characteristic of simplicial complexes, where each subset of an occurring interaction should also occur, places them outside the scope of this introduction, which is specifically focused on hypergraphs.

In this article, our focus is on model-based clustering for simple hypergraphs, specifically studying stochastic hypergraph blockmodels. We formulate a general stochastic blockmodel for simple hypergraphs, along with various submodels (Section 2.1). We provide the first result on the generic identifiability of parameters in a hypergraph stochastic blockmodel (Section 2.2). Parameter inference and node clustering are performed using a variational Expectation-Maximization (`VEM`) algorithm (Section 2.3) that approximates the maximum likelihood estimator. Model selection for the number of groups is based on an integrated classification likelihood (ICL) criterion (Section 2.4). To illustrate the performance of our method, we conduct experiments on synthetic sparse hypergraphs, including a comparison with hypergraph spectral clustering (HSC) and modularity approaches (Section 3). Notably, the line clustering experiment (Section 3.4) highlights the significant differences between our approach and the one proposed by Chodrow et al. (2021). We also analyze a co-authorship dataset, presenting conclusions that differ from spectral clustering and bipartite stochastic blockmodels (Section 4). We discuss (Section 5) our approach, its advantages, current limitations and possible extensions. An `R` package, `HyperSBM`, which implements our method in efficient `C++` code, as well as all associated scripts, are available in Appendix S1. This manuscript is accompanied by a Appendix S1 that contains additional information and experiments, as well as the proofs of all theoretical results.

2 | A STOCHASTIC BLOCKMODEL FOR HYPERGRAPHS

2.1 | Model formulation

Let $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ represent a binary hypergraph, where $\mathcal{V} = \{1,\dots,n\}$ is a set of n nodes and $\mathcal{E}$ is the set of hyperedges. In this context, a hyperedge of size $m \geq 2$ is defined as a collection of m distinct nodes from $\mathcal{V}$. We do not allow for hyperedges to be multisets or self-loops. Let $M = \max_{e\in\mathcal{E}}|e|$ denote the largest possible size of hyperedges in $\mathcal{E}$, with $M \geq 2$ (for graphs, $M = 2$). We define the sets of (unordered) node subsets, (ordered) node tuples, and hyperedges of size m as

$$\begin{aligned}
\mathcal{V}^{(m)} &= \{\{i_1,\dots,i_m\} : i_1,\dots,i_m \in \mathcal{V} \text{ are all distinct}\},\\
\mathcal{V}^{m} &= \{(i_1,\dots,i_m) : i_1,\dots,i_m \in \mathcal{V} \text{ are all distinct}\},\\
\mathcal{E}^{(m)} &= \left\{\{i_1,\dots,i_m\} \in \mathcal{V}^{(m)} : \{i_1,\dots,i_m\} \in \mathcal{E}\right\},
\end{aligned}$$

respectively. Obviously $\mathcal{E} = \bigcup_{m=2}^{M}\mathcal{E}^{(m)} \subseteq \bigcup_{m=2}^{M}\mathcal{V}^{(m)}$. For each node subset $\{i_1,\dots,i_m\} \in \mathcal{V}^{(m)}$, we define the indicator variable:

$$Y_{i_1,\dots,i_m} = \mathbb{1}_{\{i_1,\dots,i_m\}\in\mathcal{E}} = \begin{cases} 1 & \text{if } \{i_1,\dots,i_m\} \in \mathcal{E},\\ 0 & \text{if } \{i_1,\dots,i_m\} \notin \mathcal{E}.\end{cases}$$

We represent a random hypergraph as $\boldsymbol{Y} = (Y_{i_1,\dots,i_m})_{i_1,\dots,i_m\in\mathcal{V}^{(m)},2\leq m\leq M}$.

Similar to the formulation of the stochastic blockmodel (SBM) for graphs, we assume that the nodes in the hypergraph belong to Q unobserved groups. We use $Z_1,\dots,Z_n$ to denote n independent and identically distributed latent variables, where Z_i follows a prior distribution $\pi_q = \mathbb{P}(Z_i = q)$ for each $q = 1,\dots,Q$. The values π_q satisfy $\pi_q \geq 0$ and $\sum_{q=1}^{Q}\pi_q = 1$. To simplify notation, we sometimes represent Z_i as a binary vector $Z_i = (Z_{i1},\dots,Z_{iQ}) \in \{0,1\}^Q$, where only one element, Z_{iq}, equals 1. We also define $\boldsymbol{Z} = (Z_1,\dots,Z_n)$. Every m-subset of nodes $\{i_1,\dots,i_m\}$ in $\mathcal{V}^{(m)}$ is associated to a latent configuration, namely a set $\{Z_{i_1},\dots,Z_{i_m}\} = \{q_1,\dots,q_m\}$ of latent groups to which these nodes belong. The values of the latent groups within a configuration may be repeated, so that each $\{q_1,\dots,q_m\}$ is a multiset. Now, given the latent variables $\boldsymbol{Z}$, all indicator variables $Y_{i_1,\dots,i_m}$ are assumed to be independent and to follow a Bernoulli distribution whose parameter depends on the latent configuration:

$$Y_{i_1,\dots,i_m}|\{Z_{i_1} = q_1,\dots,Z_{i_m} = q_m\} \sim \mathcal{B}(B_{q_1,\dots,q_m}), \qquad \text{for any } \{i_1,\dots,i_m\} \in \mathcal{V}^{(m)}.$$

Here $B_{q_1,\dots,q_m} = B^{(m,n)}_{q_1,\dots,q_m}$ represents the probability that m unordered nodes, with latent configuration $\{q_1,\dots,q_m\}$, are connected into a hyperedge. To simplify notation, we drop the superscript (m,n). However, the model may account for two possible sparse settings. First, as the number of nodes n increases, it is natural to assume that the probability of a hyperedge may decrease; otherwise, we would only observe dense hypergraphs. Second, it is likely that real data contain fewer hyperedges of larger size m. Each B is a fully symmetric tensor of rank m, namely

$$B_{q_1,\dots,q_m} = B_{q_{\sigma(1)},\dots,q_{\sigma(m)}},\ \forall q_1,\dots,q_m \text{ and } \forall\sigma \text{ permutation of } \{1,\dots,m\}. \tag{1}$$

We denote the parameter vector as $\theta = (\pi_q, B_{q_1,\dots,q_m})_{q,m,q_1\leq\dots\leq q_m}$ and the corresponding probability distribution and expectation as $\mathbb{P}_\theta, \mathbb{E}_\theta$, respectively.

Lemma 1. *The number of different parameters in each tensor* $B = (B_{q_1,\dots,q_m})_{1\leq q_1\leq\dots\leq q_m\leq Q}$ *is* $\binom{Q+m-1}{m}$.

As a result, the total number of parameters in our hypergraph stochastic blockmodel (HSBM) is given by:

$$(Q-1) + \sum_{m=2}^{M}\binom{Q+m-1}{m}.$$

As shown in Table 1, the number of $B_{q_1,\dots,q_m}$ parameters increases rapidly as the values of Q and m grow. Note that the number of parameters (of the order $O(MQ^M + Q)$) remains small compared to the number of observations $\left(\sum_{m=2}^{M}\binom{n}{m} = O(n^M)\right)$. So we do have enough statistical information to estimate all parameters. Nonetheless, to reduce the complexity of the model, we introduce submodels by assuming equality of certain conditional probabilities $B_{q_1,\dots,q_m}$. In particular, we consider two *affiliation* submodels given by

$$B_{q_1,\dots,q_m} = \begin{cases} \alpha^{(m)} & \text{if } q_1 = \dots = q_m, \\ \beta^{(m)} & \text{if there exist at least } q_i \neq q_j \text{ for } i \neq j, \end{cases} \tag{Aff-m}$$

TABLE 1 Number $\binom{Q+m-1}{m}$ of connectivity parameters $B_{q_1,\dots,q_m}$ of the full hypergraph stochastic blockmodel for given values of Q (number of latent groups) and m (hyperedge size).

	Q					
m	**2**	**3**	**4**	**5**	**6**	**7**
3	4	10	20	35	56	84
4	5	15	35	70	126	210
5	6	21	56	126	252	462
6	7	28	84	210	462	924
7	8	36	120	330	792	1716

and

$$B_{q_1,\dots,q_m} = \begin{cases} \alpha & \text{if } q_1 = \cdots = q_m \\ \beta & \text{if there exist at least } q_i \neq q_j \text{ for } i \neq j \end{cases} \qquad \forall m = 2,\dots,M. \tag{Aff}$$

The number of parameters is dropped to $(Q-1)+2(M-1)$ and to $(Q-1)+2$ under Assumptions (**Aff-m**) and (**Aff**), respectively. These submodels align with the concepts discussed in Kamiński et al. (2019) and Chodrow et al. (2021), where they propose that only hyperedges with nodes belonging to the same group should contribute to the increase in modularity. Additionally, when $\alpha^{(m)} > \beta^{(m)}$ (resp. $\alpha > \beta$) these submodels correspond to the scenarios in which Ghoshdastidar and Dukkipati (2014, 2017) obtained their results.

A summary of the manuscript notation is given in Table 2.

2.2 | Parameter identifiability

We first establish the generic identifiability of the parameter in a HSBM that is restricted to simple m-uniform hypergraphs for any $m \geq 2$. In a parametric context, generic identifiability implies that the distribution $\mathbb{P}_\theta$ of a hypergraph over a set of n nodes uniquely defines the parameter θ, except possibly for some parameters in a subset of dimension strictly smaller than the full parameter space. In other words, if we randomly select a parameter $\theta \in \Theta$ according to the Lebesgue measure, the distribution $\mathbb{P}_\theta$ uniquely characterizes the parameter θ, for a large enough number of nodes n. Identifiability is established up to label switching on the node groups, as is common in discrete latent variable models. For the case of $m = 2$, the identifiability result corresponds to Thm. 2 in Allman et al. (2011). Our proof follows similar principles, building upon a key result by Kruskal (1977). In our case, we crucially additionally rely on a sufficient condition for a sequence of nonnegative integers to represent the degree sequence of a simple m-uniform hypergraph, as established by Behrens et al. (2013).

Theorem 1. *For any $m \geq 2$ and any integer Q, the parameter $\theta = (\pi_q, B_{q_1,\dots,q_m})_{1\leq q\leq Q, 1\leq q_1\leq\dots\leq q_m\leq Q}$ of the HSBM restricted to m-uniform simple hypergraphs over n nodes, is generically identifiable, up to label switching on the node groups, as soon as $n \geq Q^2(m!Qm + m - 1)^{2/(m-1)}$. Moreover, the result remains valid when the group proportions π_q are fixed.*

TABLE 2 Notation summary.

$\mathcal{H} = (\mathcal{V}, \mathcal{E})$	Hypergraph with $\mathcal{V} = \{1,\dots,n\}$ set of nodes and $\mathcal{E}$ collection of (simple) hyperedges
M, Q	Largest hyperedge size and number of clusters
$\mathcal{V}^{(m)}$	Node subsets (unordered) of size $2 \le m \le M$
$\mathcal{V}^m$	Node tuples (ordered) of size $2 \le m \le M$
$\mathcal{E}^{(m)}$	Hyperedges of size $2 \le m \le M$
$\boldsymbol{Y} = (Y_{i_1,\dots,i_m})_{\{i_1,\dots,i_m\}\in\mathcal{V}^{(m)},2\le m\le M}$	Observations (presence/absence of a hyperedge at each node subset)
$\boldsymbol{Z} = (Z_1,\dots,Z_n)$	Latent configurations (latent clusters), each $Z_i \in \{1,\dots,Q\}$
$\pi_q = \mathbb{P}(Z_i = q) \in [0,1]$	Clusters proportions, such that $\sum_{q=1}^{Q} \pi_q = 1$
$B_{q_1,\dots,q_m} \in [0,1]$	Probability of a hyperedge at a size-m node subset with latent configuration $\{q_1,\dots,q_m\}$, for $1 \le q_1 \le \dots \le q_m \le Q$
$\theta = (\pi_q, B_{q_1,\dots,q_m})_{q,m,q_1\le\dots\le q_m}$	Model parameter
$\alpha^{(m)}, \beta^{(m)}$ (resp. α, β)	Within-clusters and between-clusters probabilities in the affiliation sub-model (**Aff-m**) (resp. (**Aff**))
$\mathbb{Q}_\tau(\cdot)$	Variational distribution on the latent configurations $\boldsymbol{Z}$
$\tau_{iq} \in [0,1]$	Variational probability that node i belongs to cluster q, such that $\sum_{q=1}^{Q} \tau_{iq} = 1$ for all $i \in \{1,\dots,n\}$
$f(y,b)$	Bernoulli density at y with parameter b
$\mathcal{J}(\theta,\tau)$	Evidence lower bound (ELBO)
$\mathcal{H}(\cdot), \mathrm{KL}(\cdot\|\|\cdot)$	Entropy and Kullback–Leibler divergence

This result does not provide specific insights into the identifiability in the affiliation cases (**Aff-m**) and (**Aff**). Indeed, it does not explicitly characterize the subspace of the parameter space where identifiability may not hold, although we know that its dimension is smaller than that of the full parameter space (and that the possible restrictions apply only on the part of the parameter space concerning the probabilities of connection $B_{q_1,\dots,q_m}$).

The result we have established for m-uniform hypergraphs also implies a similar result for non-uniform simple hypergraphs, as shown in the following corollary.

> **Corollary 1.** *For any integer Q, the parameter $\theta = (\pi_q, B_{q_1,\dots,q_m})_{1\le q\le Q, 1\le q_1\le\dots\le q_m\le Q, 2\le m\le M}$ of the HSBM for simple hypergraphs over n nodes is generically identifiable, up to label switching on the node groups, as soon as $n \ge Q^2(M!QM + M - 1)^{2/(M-1)}$.*

Our proof of Corollary 1 relies on the assumption that all the π_q's are distinct, which is a generic condition. This condition is not explicitly stated in the corollary, but it is required for the proof to hold. Consequently, the result of generic identifiability does not bring any insight in cases where the group proportions are equal, as it is not sufficient to identify the parameters separately for each value of m.

Additional technical work is thus needed to establish whether a HSBM with equal group proportions, or whether the affiliation submodels have identifiable parameters.

As a final note, we mention that there is no direct link between parameter identifiability and detectability thresholds for clusters recovery (Dumitriu et al., 2022; Stephan & Zhu, 2022). While

clusters recovery is an asymptotic result with guarantees when the sample size increases, parameter identifiability is a theoretical result stating that the distribution of the observations (for a minimal sample size) fully characterizes the parameter. It is theoretical in the sense that it does not deal with inference, though the property has consequences on inference results. Parameter identifiability is a basic requirement for consistency results of maximum likelihood estimators to hold in parametric settings and it is also required for proofs of clusters exact recovery.

2.3 | Parameter estimation via variational expectation-maximization

The likelihood of the model is given as a marginal distribution

$$\begin{aligned}\mathbb{P}_\theta(\boldsymbol{Y}) &= \sum_{q_1=1}^{Q}\cdots\sum_{q_n=1}^{Q}\mathbb{P}_\theta(\boldsymbol{Y},\boldsymbol{Z}=(q_1,\ldots,q_n))\\ &= \sum_{q_1=1}^{Q}\cdots\sum_{q_n=1}^{Q}\left(\prod_{i=1}^{n}\mathbb{P}_\theta(Z_i=q_i)\right)\prod_{m=2}^{M}\prod_{\{i_1,\ldots,i_m\}\in\mathcal{V}^{(m)}}\mathbb{P}_\theta(Y_{i_1,\ldots,i_m}|Z_{i_1}=q_{i_1},\ldots,Z_{i_m}=q_{i_m})\\ &= \sum_{q_1=1}^{Q}\cdots\sum_{q_n=1}^{Q}\left(\prod_{i=1}^{n}\pi_{q_i}\right)\prod_{m=2}^{M}\prod_{\{i_1,\ldots,i_m\}\in\mathcal{V}^{(m)}}(B_{q_{i_1},\ldots,q_{i_m}})^{Y_{i_1\ldots i_m}}(1-B_{q_{i_1},\ldots,q_{i_m}})^{1-Y_{i_1\ldots i_m}}. \end{aligned} \tag{2}$$

The computation of the model likelihood is generally intractable. Equation (2) involves a summation over all possible Q^n different latent configurations of the nodes, which becomes computationally prohibitive when n and Q are large. In the context of latent variable models, the Expectation-Maximization (EM) algorithm (Dempster et al., 1977) is commonly used to address this issue. However, the EM algorithm cannot be directly applied to SBMs. This is because the E-step, which involves computing the conditional posterior distribution of the latent variables $P(\boldsymbol{Z}|\boldsymbol{Y})$, is itself intractable in SBMs (see, e.g., Matias & Robin, 2014). One possible solution is to employ variational approximations of the EM algorithm, known as Variational EM (VEM, Jordan et al., 1999). Below, we recall the classical approach for the VEM algorithm.

We denote the density of the Bernoulli distribution with parameter b as

$$\forall y\in\{0,1\},\quad f(y,b) := y\log b+(1-y)\log(1-b). \tag{3}$$

Then, the complete data log-likelihood is

$$\begin{aligned}\ell_n^c(\theta) &= \log\mathbb{P}_\theta(\boldsymbol{Y},\boldsymbol{Z}) = \log\mathbb{P}_\theta(\boldsymbol{Z})+\log\mathbb{P}_\theta(\boldsymbol{Y}|\boldsymbol{Z})\\ &= \sum_{q=1}^{Q}\sum_{i=1}^{n}Z_{iq}\log\pi_q+\sum_{m=2}^{M}\sum_{q_1=1}^{Q}\cdots\sum_{q_m=1}^{Q}\sum_{\{i_1,\ldots,i_m\}\in\mathcal{V}^{(m)}}Z_{i_1q_1}\cdots Z_{i_mq_m}f(Y_{i_1\ldots i_m},B_{q_1,\ldots,q_m})\\ &= \sum_{q=1}^{Q}\sum_{i=1}^{n}Z_{iq}\log\pi_q+\sum_{m=2}^{M}\sum_{q_1\le q_2\le\ldots\le q_m}\sum_{(i_1,\ldots,i_m)\in\mathcal{V}^m}Z_{i_1q_1}\cdots Z_{i_mq_m}f(Y_{i_1\ldots i_m},B_{q_1,\ldots,q_m}). \end{aligned} \tag{4}$$

Note that the final equality ensures that each parameter value appears only once. The key principle underlying the variational method is to adopt the same iterative two-step structure as the EM algorithm but replace the intractable posterior distribution $\mathbb{P}_\theta(\boldsymbol{Z}|\boldsymbol{Y})$ with the best approximation, in terms of Kullback–Leibler divergence, from a class of simpler distributions, often factorized.

We introduce a class of probability distributions $\mathbb{Q}_\tau$ over $\boldsymbol{Z} = (Z_1, \ldots, Z_n)$ that factorize over the set of nodes, thus given by

$$\mathbb{Q}_\tau(\boldsymbol{Z}) = \prod_{i=1}^{n} \mathbb{Q}_\tau(Z_i) = \prod_{i=1}^{n} \prod_{q=1}^{Q} \tau_{iq}^{Z_{iq}},$$

where the variational parameter $\tau_{iq} = \mathbb{Q}_\tau(Z_i = q) \in [0,1]$ satisfies $\sum_{q=1}^{Q} \tau_{iq} = 1$ for any $i = 1, \ldots, n$. The expectation under distribution $\mathbb{Q}_\tau$ is denoted as $\mathbb{E}_{\mathbb{Q}_\tau}$, and $\mathcal{H}(\mathbb{Q}_\tau)$ represents the entropy of $\mathbb{Q}_\tau$. Now we define the evidence lower bound (ELBO):

$$\begin{aligned}
\mathcal{J}(\theta, \tau) &= \mathbb{E}_{\mathbb{Q}_\tau}[\log \mathbb{P}_\theta(\boldsymbol{Y}, \boldsymbol{Z})] + \mathcal{H}(\mathbb{Q}_\tau) \\
&= \mathbb{E}_{\mathbb{Q}_\tau}[\log \mathbb{P}_\theta(\boldsymbol{Y}, \boldsymbol{Z})] - \mathbb{E}_{\mathbb{Q}_\tau}[\log \mathbb{Q}_\tau(\boldsymbol{Z})] \\
&= \sum_{q=1}^{Q} \sum_{i=1}^{n} \tau_{iq} \log \frac{\pi_q}{\tau_{iq}} + \sum_{m=2}^{M} \sum_{q_1 \leq q_2 \leq \ldots \leq q_m} \sum_{(i_1, \ldots, i_m) \in \mathcal{V}^m} \tau_{i_1 q_1} \cdots \tau_{i_m q_m} f(Y_{i_1 \ldots i_m}, B_{q_1, \ldots, q_m}).
\end{aligned} \tag{5}$$

It can be observed that $\mathcal{J}(\theta, \tau)$ satisfies the following relation:

$$\mathcal{J}(\theta, \tau) = \log \mathbb{P}_\theta(\boldsymbol{Y}) - \mathrm{KL}(\mathbb{Q}_\tau(\boldsymbol{Z}) || \mathbb{P}_\theta(\boldsymbol{Z}|\boldsymbol{Y})), \tag{6}$$

where $\mathrm{KL}(\cdot||\cdot)$ denotes the Kullback–Leibler divergence. Equation (6) is at the core of the EM algorithm and its variational approximation. In the classical EM approach, at the tth iteration of the algorithm, the variational distribution $\mathbb{Q}_\tau$ is chosen as the distribution $\mathbb{P}_{\theta^{(t)}}(\boldsymbol{Z}|\boldsymbol{Y})$ of the latent variables given the observations at the current parameter value $\theta^{(t)}$. This cancels the Kullback–Leibler term and the ELBO equals the log-likelihood. When the distribution $\mathbb{P}_\theta(\boldsymbol{Z}|\boldsymbol{Y})$ is not factorized, such a choice would prevent from an efficient computation of the expectation $\mathbb{E}_{\mathbb{Q}_\tau}[\log \mathbb{P}_\theta(\boldsymbol{Y}, \boldsymbol{Z})]$ of the complete log-likelihood under $\mathbb{Q}_\tau$ appearing in Equation (5). Thus, the variational approximation searches for the best approximation of the true $\mathbb{P}_\theta(\boldsymbol{Z}|\boldsymbol{Y})$ in a class of simplified (in general, factorized) variational distributions. As a consequence, the Kullback–Leibler divergence term in (6) is nonnull in general and the ELBO $\mathcal{J}$ serves as a lower bound for the model log-likelihood $\log \mathbb{P}_\theta(\boldsymbol{Y})$. The VEM algorithm iterates between the following two steps until a suitable convergence criterion is met:

- VE-**Step** maximizes $\mathcal{J}(\theta, \tau)$ with respect to τ

$$\widehat{\tau}^{(t)} = \arg\max_{\tau} \mathcal{J}(\theta^{(t-1)}, \tau), \quad \text{s.t.} \sum_{q=1}^{Q} \widehat{\tau}_{iq}^{(t)} = 1 \qquad \forall i = 1, \ldots, n. \tag{7}$$

  This step involves finding the best approximation of the conditional distribution $\mathbb{P}_\theta(\boldsymbol{Z}|\boldsymbol{Y})$ by minimizing the Kullback–Leibler divergence term in (6).

- M-**Step** maximizes $\mathcal{J}(\theta, \tau)$ with respect to θ

$$\widehat{\theta}^{(t)} = \arg\max_{\theta} \mathcal{J}(\theta, \tau^{(t-1)}), \quad \text{s.t.} \sum_{q=1}^{Q} \widehat{\pi}_q^{(t)} = 1. \tag{8}$$

  This step updates the values of the model parameters π_q and $B_{q_1, \ldots, q_m}$.

In the following we provide the solutions of the two maximization problems in Equations (7) and (8).

Proposition 1 (VE-Step). *Given the current model parameters* $\theta = (\pi_q, B_{q_1,..,q_m})_{q,m,q_1\leq..\leq q_m}$ *at any iteration of the* VEM *algorithm, the corresponding optimal values of the variational parameters* $(\widehat{\tau}_{iq})_{i,q}$ *defined in Equation (7) satisfy the fixed point equation*:

$$\log \widehat{\tau}_{iq} = \log \pi_q + \sum_{m=1}^{M-1} \sum_{1\leq q_1\leq..\leq q_m\leq Q} \sum_{\substack{(i_1,\ldots,i_m)\in\mathcal{V}^m \\ \text{s.t.}\{i,i_1,\ldots,i_m\}\in\mathcal{V}^{(m+1)}}} \widehat{\tau}_{i_1 q_1}\cdots\widehat{\tau}_{i_m q_m} f\left(Y_{i i_1\ldots i_m}, B^{(m+1,n)}_{q q_1..q_m}\right) + c_i, \quad (9)$$

for any $1 \leq i \leq n$ *and* $1 \leq q \leq Q$ *and where* c_i *are normalizing constants such that* $\sum_q \widehat{\tau}_{iq} = 1$.

Equation (9) relates the variational probability $\widehat{\tau}_{iq}$ that a node i belongs to a cluster q to the other variational parameters (as well as the observations and current parameter value θ). The sum starts at $m = 1$ and deals with $(m + 1)$-tuples of nodes $\{i, i_1,\ldots, i_m\}$ that contain node i and whose latent configuration is given by some multiset $\{q, q_1,\ldots, q_m\}$.

Remark 1. From Proposition 1, the τ_i's are obtained using a fixed point algorithm. Although in all the situations we experienced, the algorithm converged in a reasonable number of iterations, we have no guarantee about existence nor uniqueness of a solution to (9).

Proposition 2 (M-Step). *Given the variational parameters* $(\tau_{iq})_{i,q}$ *at any iteration of the* VEM *algorithm, the corresponding optimal values of the model parameters* $(\widehat{\pi}_q, \widehat{B}_{q_1..q_m})_{q,m,q_1\leq..\leq q_m}$ *defined in Equation (8) are given by*

$$\widehat{\pi}_q = \frac{1}{n}\sum_{i=1}^{n}\tau_{iq} \qquad \text{and} \qquad \widehat{B}_{q_1\ldots q_m} = \frac{\sum_{(i_1,\ldots,i_m)\in\mathcal{V}^m} \tau_{i_1 q_1}\ldots\tau_{i_m q_m} Y_{i_1\ldots i_m}}{\sum_{(i_1,\ldots,i_m)\in\mathcal{V}^m} \tau_{i_1 q_1}\ldots\tau_{i_m q_m}}.$$

We now express the solutions of the M-Step under the submodels given by (**Aff-m**) and (**Aff**). Note that the VE-Step is unchanged under these settings.

Proposition 3 (M-Step, affiliation setups). *In the particular affiliation submodels given by (***Aff-m***) and (***Aff***), respectively, given variational parameters* $(\tau_{iq})_{i,q}$, *at any iteration of the* VEM *algorithm, the corresponding optimal values of* $(\widehat{\alpha}^{(m)}, \widehat{\beta}^{(m)})_m$ *and* $\widehat{\alpha}, \widehat{\beta}$ *maximizing* $\mathcal{J}$ *as in Equation (8) are given by*

- *Under Assumption (***Aff-m***),*

$$\widehat{\alpha}^{(m)} = \frac{\sum_{q=1}^{Q}\sum_{i_1<\cdots<i_m}\tau_{i_1 q}\ldots\tau_{i_m q}Y_{i_1..i_m}}{\sum_{q=1}^{Q}\sum_{i_1<\cdots<i_m}\tau_{i_1 q}\ldots\tau_{i_m q}},$$

$$\widehat{\beta}^{(m)} = \frac{\sum_{\substack{q_1\leq..\leq q_m \\ |\{q_1,\ldots,q_m\}|\geq 2}}\sum_{(i_1,\ldots,i_m)\in\mathcal{V}^m}\tau_{i_1 q_1}\ldots\tau_{i_m q_m}Y_{i_1..i_m}}{\sum_{\substack{q_1\leq..\leq q_m \\ |\{q_1,\ldots,q_m\}|\geq 2}}\sum_{(i_1,\ldots,i_m)\in\mathcal{V}^m}\tau_{i_1 q_1}\ldots\tau_{i_m q_m}}.$$

- *Under Assumption (**Aff**),*

$$\widehat{\alpha} = \frac{\sum_{m=2}^{M}\sum_{q=1}^{Q}\sum_{i_1<\cdots<i_m}\tau_{i_1q}\ldots\tau_{i_mq}Y_{i_1..i_m}}{\sum_{m=2}^{M}\sum_{q=1}^{Q}\sum_{i_1<\cdots<i_m}\tau_{i_1q}\ldots\tau_{i_mq}},$$

$$\widehat{\beta} = \frac{\sum_{m=2}^{M}\sum_{\substack{q_1\leq..\leq q_m\\ |\{q_1,\ldots,q_m\}|\geq 2}}\sum_{(i_1,\ldots,i_m)\in\mathcal{V}^m}\tau_{i_1q_1}\ldots\tau_{i_mq_m}Y_{i_1..i_m}}{\sum_{m=2}^{M}\sum_{\substack{q_1\leq..\leq q_m\\ |\{q_1,\ldots,q_m\}|\geq 2}}\sum_{(i_1,\ldots,i_m)\in\mathcal{V}^m}\tau_{i_1q_1}\ldots\tau_{i_mq_m}}.$$

2.3.1 | Algorithm initialization

We choose to begin the algorithm with its M-step, which requires an initial value for $\boldsymbol{\tau}$. This allows us to leverage smart initialization strategies based on a preliminary clustering of the nodes. Specifically, we employ three different initialization strategies and select the best result that maximizes the ELBO criterion $\mathcal{J}$:

Random initialization
This naive method involves drawing each $(\tau_{iq})_{1\leq q\leq Q}$ uniformly from $(0,1)$ for every node i and normalizing the vector τ_i.

"Soft" spectral clustering
We utilize Alg. 1 from Ghoshdastidar and Dukkipati (2017) combined with soft k-means. In this approach, we compute a hypergraph Laplacian and construct the column matrix X consisting of its leading Q orthonormal eigenvectors. The rows of X are then normalized to have unit norm (following steps 1–3 in Alg. 1 from Ghoshdastidar & Dukkipati, 2017). We subsequently perform a soft k-means algorithm on the rows of X to obtain τ_{iq}, which represents the posterior probability of node i belonging to cluster q.

Graph-component absolute spectral clustering
This strategy focuses on edges in the hypergraph ($m = 2$) and the corresponding adjacency matrix. We apply the absolute spectral clustering method (Rohe et al., 2011) to this adjacency matrix. The absolute spectral clustering method introduces a graph Laplacian with both positive and negative eigenvalues and focuses on the ones with largest magnitude, thus capturing both communities and dis-assortative structures. It should be noted that this initialization only uses information from hyperedges of size $m = 2$, excluding hyperedges with larger sizes. However, absolute spectral clustering is considered superior to spectral clustering as it captures disassortative groups.

In Section F.2 from the Appendix S1, we include a comparison of different initialization strategies. In general, there would not be an initialization strategy that is always superior, so we recommend always using different strategies and selecting the best criteria.

2.3.2 | Fixed point

The VE-Step is achieved through a fixed-point algorithm. In practice, at iteration t of the VEM algorithm, starting from the previous values of the variational and model parameters $\tau_{iq}^{(t-1)}$

and $\theta^{(t-1)}$, respectively, we iterate over some index u to compute the values of $\tau_{iq}^{(t,u)}$ according to Equation (9). This generates a sequence of values $\tau_{iq}^{(t,u)}$. We terminate these iterations either when we reach the maximum number of fixed-point iterations ($u > U_{\max}$) or when the variational parameters have converged ($\max_{iq} |\tau_{iq}^{(t,u-1)} - \tau_{iq}^{(t,u)}| \leq \varepsilon$), where ε is a small tolerance threshold.

2.3.3 | Stopping criteria

The iterations of the VEM algorithm should be terminated when the ELBO $\mathcal{J}$ and the sequence of model parameter vectors $\theta^{(t)} = (\theta_s^{(t)})_s$ have converged. However, in practice, we have observed that sometimes the algorithm stops prematurely when the VE-Step still requires a few iterations to reach a fixed point. In such cases, continuing with the VEM iterations often leads to higher values of the ELBO function and better parameter estimates. To address this, we enforce the condition that the fixed point in the VE-Step is reached in its first iteration. This reduces the chance of converging to local maxima of $\mathcal{J}$. If these convergence conditions are not met, we stop the algorithm if the maximum number of iterations has been reached. To summarize, we stop the algorithm whenever:

$$\left\{ \frac{|\mathcal{J}(\theta^{(t-1)}) - \mathcal{J}(\theta^{(t)})|}{|\mathcal{J}(\theta^{(t)})|} \leq \varepsilon \quad \text{and} \quad \max_s |\theta_s^{(t-1)} - \theta_s^{(t)}| \leq \varepsilon \quad \text{and} \quad \max_{iq} |\tau_{iq}^{(t,0)} - \tau_{iq}^{(t,1)}| \leq \varepsilon \right\}$$
$$\text{or} \quad \{t > T_{\max}\}.$$

Section E in Appendix S1 contains additional details about the algorithm's implementation.

2.3.4 | Algorithm complexity and choice of M

The complexity of our algorithm is of the order $O\left(nQ^M \binom{n}{M}\right)$, which can be quite prohibitive for large datasets, especially when M becomes large. It is important to emphasize that when analyzing a dataset, the value of M is not necessarily the maximum observed size of the hyperedges, but rather a modeling choice. Indeed, while an occurring hyperedge $Y_{i_1,\ldots,i_m}$ with node clusters $\{q_1,\ldots,q_m\}$ contributes $\log B_{q_1,\ldots,q_m}$ to the likelihood, a non occurring one contributes $\log(1 - B_{q_1,\ldots,q_m})$ and the statistical information that they bring to the parameter is the same (see Equations (3) and (4)). Now let's consider for, for example, a co-authorship dataset where we observe n authors and at most 3 co-authors per paper. The absence of hyperedges of size 4 provides as much information for a HSBM as if all possible size-4 hyperedges were present. Similarly, the information contained in a dataset with all but five possible size-4 hyperedges present is the same as the information contained in a dataset with only five occurring size-4 hyperedges. In other words, 0 and 1 values play a symmetric role.

As a consequence, the choice of M is left to the discretion of the statistician, depending on the characteristics of the dataset and the available computational resources. In particular, if there are hyperedges with very large sizes M, the statistician may decide not to consider them, just as it is justified not to take into account the absence of hyperedges of size $M + 1$, where M is the largest observed size. It is important to note that choosing $M > 2$ already represents an improvement in terms of considering more information compared to a graph analysis of the data.

Therefore, for large datasets, we recommend limiting the analysis to smaller values of M, such as $M = 3$ or $M = 4$, to reduce computational burden and improve efficiency.

2.4 | Model selection

While Ghoshdastidar and Dukkipati (2017) propose a method for selecting the number of groups based on the spectral gap, our approach relies on a statistical framework to construct a model selection criterion.

After obtaining the estimated parameters $\hat{\theta}$ and $(\hat{\tau}_i)_i$ from the `VEM` algorithm, we assign each node i to its estimated group $\hat{Z}_i = \arg\max_q \hat{\tau}_{iq}$. We then define the Integrated Classification Likelihood (ICL, Biernacki et al., 2000) for the full model and the submodels (**Aff-m**) and (**Aff**) as follows:

$$\mathrm{ICL}_{\text{full}}(q) = \log \mathbb{P}_{\hat{\theta}}(\boldsymbol{Y}, \hat{\boldsymbol{Z}}) - \frac{1}{2}(q-1)\log n - \frac{1}{2}\sum_{m=2}^{M} \binom{q+m-1}{m} \log\binom{n}{m},$$

$$\mathrm{ICL}_{\text{aff-m}}(q) = \log \mathbb{P}_{\hat{\theta}}(\boldsymbol{Y}, \hat{\boldsymbol{Z}}) - \frac{1}{2}(q-1)\log n - (M-1)\sum_{m=2}^{M} \log\binom{n}{m},$$

$$\mathrm{ICL}_{\text{aff}}(q) = \log \mathbb{P}_{\hat{\theta}}(\boldsymbol{Y}, \hat{\boldsymbol{Z}}) - \frac{1}{2}(q-1)\log n - \sum_{m=2}^{M} \log\binom{n}{m}.$$

These criteria are constructed as the complete log-likelihood (computed at the estimated parameter values and clusters), penalized by a BIC-like term that accounts for the number of parameters and the corresponding "effective" sample size (n for the parameters related to the nodes and $\binom{n}{m}$ for the size-m interaction parameters $B_{q_1,\ldots,q_m}$). ICL criteria have been widely used in the context of SBMs. Their theoretical properties have never been established, though they exhibit very good empirical results on synthetic SBMs datasets (e.g. Daudin et al., 2008). Recently, Cerqueira and Leonardi (2020) obtained a first consistency result for a related criterion in the graph SBM, relying on a penalized version of the exact ICL (Côme & Latouche, 2015), also known in the information theory literature as Krichevsky–Trofimov (KT) estimator. While the literature of order estimation focuses on minimal penalties as these will lead to minimum underestimation probability (see for e.g. van Handel, 2011), these KT penalties are generally heavier than what is thought to be sufficient to consistently estimate the number of groups. We determine the number of groups $\hat{q}$ as the value that maximizes the corresponding ICL criterion: $\hat{q} = \arg\max_q \mathrm{ICL}(q)$.

3 | SYNTHETIC EXPERIMENTS

3.1 | Synthetic data

We conducted a simulation study to evaluate the performance of the `HyperSBM` package. We generated hypergraphs under the HSBM with two or three latent groups ($Q = 2$ or $Q = 3$). The group proportions were non-uniform, with $\pi = (0.6, 0.4)$ for $Q = 2$ and $\pi = (0.4, 0.3, 0.3)$ for $Q = 3$. We set the largest hyperedge size M to 3, and we considered different numbers of nodes, $n \in \{50, 100, 150, 200\}$.

To simplify the latent structure, we assumed the (**Aff-m**) submodel, and we parametrized the model through the ratios $\rho^{(m)}$ of within-group size-m hyperedges over between-groups size-m hyperedges (assumed constant with n, see Section F.1 in Appendix S1 for details). We analyzed two different scenarios:

A. Communities: in this scenario, we focus on community detection and consider the case of more within-group than between-groups size-m hyperedges $\rho^{(m)} > 1$ for $m = 2, 3$.
B. Disassortative: in this scenario, we focus on disassortative behaviour and consider the case of less within-group than between-groups size-m hyperedges $\rho^{(m)} < 1$ for $m = 2, 3$.

Each setting is a combination of a scenario ($X = A,B$) and number of groups ($Q = 2, 3$) and is denoted XQ. In each setting, values of $\alpha^{(m)} = \alpha^{(m,n)}$ and $\beta^{(m)} = \alpha^{(m,n)}$ (we here emphasize the dependence on the number of nodes n) decrease with increasing n so as to maintaining constant the quantities $n\alpha^{(2,n)}$ and $n\beta^{(2,n)}$ as well as $n^2\alpha^{(3,n)}$ and $n^2\beta^{(3,n)}$. This implies that the number of size-m hyperedges (both within and between groups) grows linearly with n. We have explored a total of 5 different settings, denoted by A2, A3, B2, B3, and A3' and we present below the most striking results. In the case of scenarios A (communities) with $Q = 3$ groups, we pushed the limits and explore two different settings (namely settings A3 and A3'), with setting A3 being *highly sparse*, that is, sparser than the already sparse setting A3'. Details of the parametrization, specific parameter values and number of hyperedges are fully given in Section F.1 in Appendix S1, while Section F.2 in Appendix S1 contains additional results.

For each setting and each value of n, we randomly draw 50 different hypergraphs. We estimate the parameters using the full HSBM formulation with our `VEM` algorithm, relying on soft spectral clustering (for Scenario A) and graph-component absolute spectral clustering (for Scenario B) initializations (see paragraph "Algorithm initialization" above).

3.2 | Clustering and estimation under HSBM with a fixed number of groups

In this part, we focus on clustering and parameter estimation with a known number of groups. The performance of `HyperSBM` is evaluated based on its ability to accurately recover the true clustering and estimate the original parameters. We also compare our results with hypergraph spectral clustering, relying on Alg. 1 from Ghoshdastidar and Dukkipati (2017), denoted HSC below.

3.2.1 | Clustering results

The performance of correct classification is evaluated using the Adjusted Rand Index (ARI, Hubert & Arabie, 1985). The ARI measures the similarity between the true node clustering and the estimated clustering. It is upper bounded by 1, where a value of 1 indicates perfect agreement between the clusterings, and negative values indicate less agreement than expected by chance.

Figure 1 displays the boxplot values of the ARI for settings A2 to B3. It is evident that our `HyperSBM` consistently outperforms HSC, obtaining higher ARI values overall and significantly lower variances in most cases, except for setting B3, where `HyperSBM` exhibits a larger variance

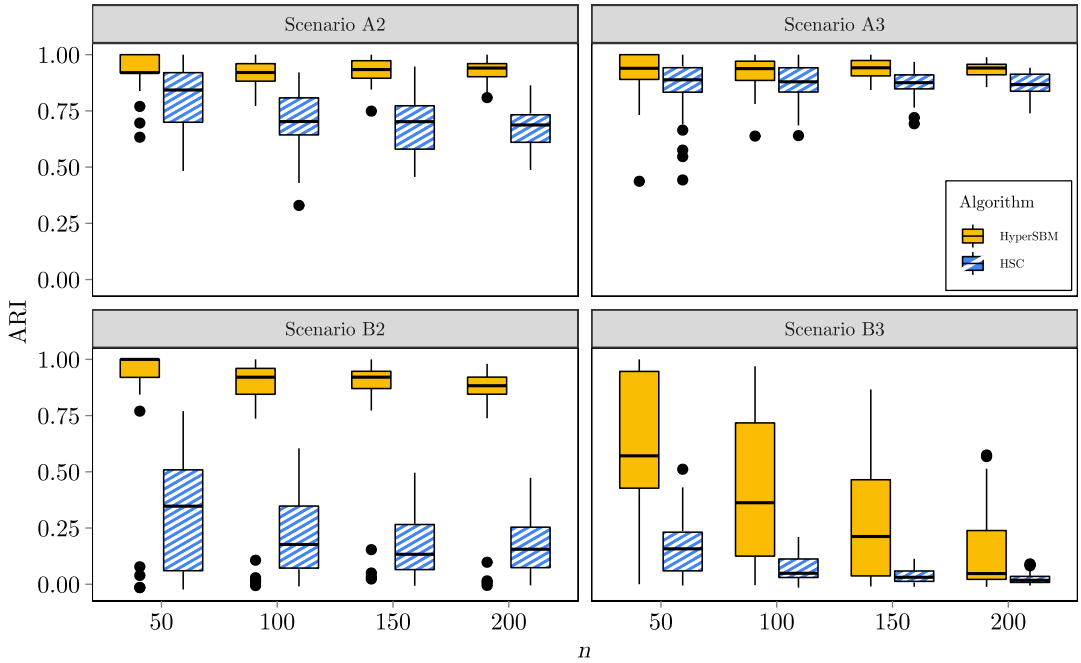

FIGURE 1 Boxplots of adjusted Rand indexes for different settings XQ (where $X = A, B$ is the scenario and $Q = 2, 3$ is the number of groups), number of nodes n (along x-axis) and two methods: our `HyperSBM` (left boxplot) and HSC (right boxplot). First row (resp. first column) shows scenario A with communities (resp. $Q = 2$) while second row (resp. second column) shows scenario B with disassortative behavior (resp. $Q = 3$).

but still yields substantially better results compared to HSC. We also observe that increasing the number of nodes n does not appear to significantly enhance the clustering results of `HyperSBM`. This behavior could be attributed to our simulation setting, where the numbers of size-m hyperedges ($m = 2, 3$) are kept linearly increasing with n. However, it is worth noting that the variances of the ARI obtained by `HyperSBM` tend to decrease with an increasing number of nodes n. One final comment pertains to the relatively poor clustering performance obtained by both methods in setting B3: this setting appears to be particularly challenging.

3.2.2 | Parameter estimation accuracy

We also evaluate the accuracy of parameter estimation. As the parameter values may be extremely small (see Section F.1 in Appendix S1), we choose to compute the Mean Squared Relative Error (MSRE) between the true parameters (in the full model) and the estimated values, both for the prior probabilities π_q and the probabilities of hyperedge occurrence $B^{(m)}_{q_1,\dots,q_m}$. Specifically, we compute the aggregated MSRE over all the components of θ using the following formula:

$$\mathrm{MSRE} = \frac{1}{n_{rep}} \sum_{i=1}^{n_{rep}} \left\{ \sum_{q=1}^{Q-1} \left(\frac{\hat{\pi}^i_q - \pi_q}{\pi_q} \right)^2 + \sum_{m=2}^{M} \sum_{q_1 \leq \dots \leq q_m} \left(\frac{\hat{B}^i_{q_1,\dots,q_m} - B_{q_1,\dots,q_m}}{B_{q_1,\dots,q_m}} \right)^2 \right\},$$

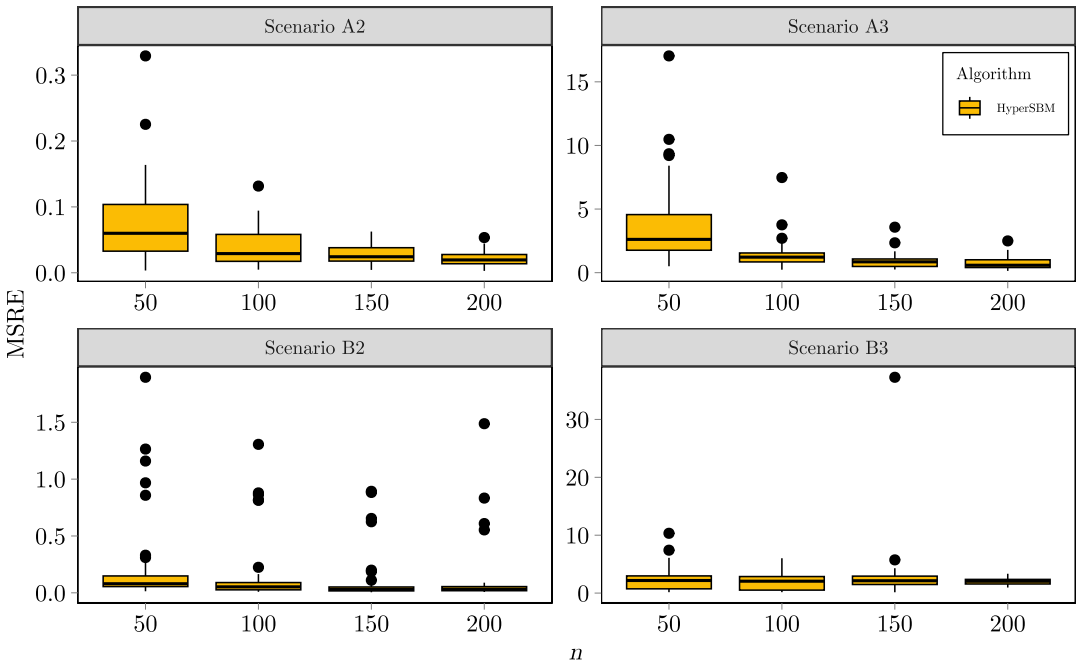

FIGURE 2 Boxplots of Mean Squared Relative Errors between true and estimated model parameters for different settings XQ (where $X = A, B$ is the scenario and $Q = 2, 3$ is the number of groups) and number of nodes n (along x-axis). First row (resp. first column) shows scenario A with communities (resp. $Q = 2$) while second row (resp. second column) shows scenario B with disassortative behavior (resp. $Q = 3$).

where $(\hat{\pi}^i_1, \ldots, \hat{\pi}^i_{Q-1}, \{\hat{B}^i_{q_1,\ldots,q_m}\}_{m,q_1,\ldots,q_m})$ is the set of free parameters estimated on the i-th dataset by the full model and $n_{rep} = 50$ is the number of replicates.

The corresponding results are summarized through the boxplots in Figure 2. The relative errors are rather small, decreasing and showing a lower variance as the number of nodes increases. Note that the absolute values of MSRE cannot be compared between the cases $Q = 2$ (first column) and $Q = 3$ (second column), with very different scales on the y-axis. Indeed, in the first case, the relative error is cumulated over a total of $1 + 3 + 4 = 8$ free parameters (in the full model), while this increases to $2 + 6 + 10 = 18$ free parameters when $Q = 3$.

3.3 | Performance of model selection

In this section we assess the performance of ICL as a model selection criterion. The simulated data is fitted with our `HyperSBM` with a number of latent states ranging from 1 to 5.

In Table 3, we show the frequency of the selected number of groups for setting A3'. The correct model is selected in 74% of cases for $n = 50$, in 98% of cases for $n = 100$ and in 100% of cases for $n = 150, 200$. We also compute the value of ARI of the classification obtained with three clusters depending on the selected number of latent groups. This value is always equal to 1 when the correct model is recovered, thus confirming the optimal behavior of `HyperSBM` already shown in Section 3.2.

TABLE 3 Frequency (as a percentage) of the selected number of groups Q for setting A3'.

Q	$n = 50$	$n = 100$	$n = 150$	$n = 200$
1	0%	0%	0%	0%
2	26%	2%	0%	0%
3	74%	98%	100%	100%
4	0%	0%	0%	0%
5	0%	0%	0%	0%

Note: Model selection is carried out by means of the ICL criterion. Results are computed over 50 samples for each value of n.

TABLE 4 Description settings for the line clustering experiments.

	Number of points per line	Number of noisy points	Total number of points	Mean number of hyperedges
Two lines	30	40	100	1070.84
Three lines	30	60	150	587.70

3.4 | Line clustering through hypergraphs

Following Leordeanu and Sminchisescu (2012); Kamiński et al. (2019) and earlier references, we here explore the use of hypergraphs to detect line clusters of points in $\mathbb{R}^2$. Similarly to the construction of pairwise similarity measures, we here resort on third-order affinity measures to detect alignment of points since pairwise measures would be useless to detect alignment. Thus, for any triplet of points $\{i, j, k\}$, we use the mean distance to the best fitting line as a dissimilarity measure $d(i, j, k)$ and transform this through a Gaussian kernel to a similarity measure.

We performed two different experiments, with either two or three lines. In each setting, we randomly generate the same number of points per line in the range $[-0.5, 0.5]^2$ and perturbed with centered Gaussian noise with standard deviation equal to 0.01. We then add noisy points, generated from uniform distribution on $[-0.5, 0.5]^2$. The particular settings of each experiment are described in Table 4 and Figure 3 shows the resulting sets of points.

For both settings, we generated 100 different 3-uniform hypergraphs using the following procedure. We randomly selected 3 points $\{i, j, k\}$ and calculated the mean distance $d(i, j, k)$ to the best-fitting line. We then measured their similarity using a Gaussian kernel $\exp(-d(i, j, k)^2/\sigma^2)$ with $\sigma^2 = 0.04$. If the similarity was greater than a threshold $\epsilon = 0.999$, we constructed a hyperedge $\{i, j, k\}$. This procedure resulted in both signal hyperedges, where all points belonged to the same line cluster, and noise hyperedges, where the points were sufficiently aligned without belonging to the same line. The signal-to-noise ratio of hyperedges was set to 2 for each hypergraph. We specifically simulated sparse hypergraphs, and the average number of hyperedges is presented in Table 4. Additionally, isolated nodes in the hypergraph were excluded from the clustering analysis.

We applied our `HyperSBM` algorithm to cluster the nodes of these 3-uniform and sparse hypergraphs, and we compared the results with three different modularity-based approaches. The first two approaches, referred to as Chodrow_Symm and Chodrow_AON, are from Chodrow et al. (2021) and are based on their general symmetric and all-or-nothing modularity, respectively. The third approach, referred to as Kaminski, is from Kamiński et al. (2019). The modularity-based

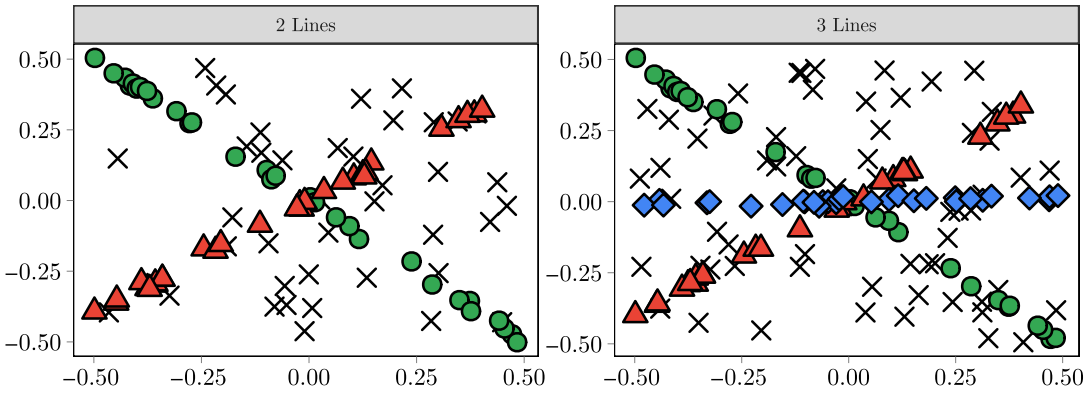

FIGURE 3 Sets of points from the line clustering experiments. Left: two lines (green dots and red triangles) plus noise (black crosses). Right: three lines (green dots, red triangles and blue diamonds) plus noise (black crosses).

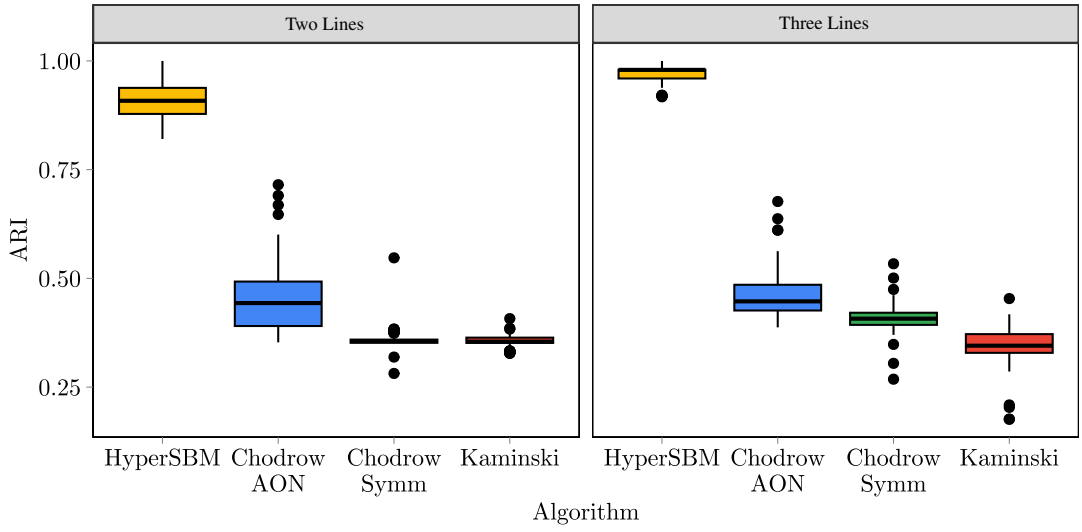

FIGURE 4 Boxplots of the adjusted Rand index obtained by the different clustering methods on the line clustering problem. Left: two lines, right: three lines.

methods automatically select the number of groups, and for `HyperSBM`, we performed model selection using $Q \in \{1,\ldots,6\}$.

Figure 4 displays the ARI obtained from the clustering results. We can observe that the modularity-based methods fail to accurately recover the true original line clusters, resulting in lower ARIs. In contrast, `HyperSBM` shows good performance in this task, achieving higher ARIs. This difference in performance can be attributed to the tendency of modularity-based methods, especially the one by Kamiński et al. (2019), to select a larger number of groups in this particular context, as evidenced in Figure 5.

This experiment highlights the distinct behavior of `HyperSBM` compared to the modularity-based clustering methods, including the approach proposed by Chodrow et al. (2021),

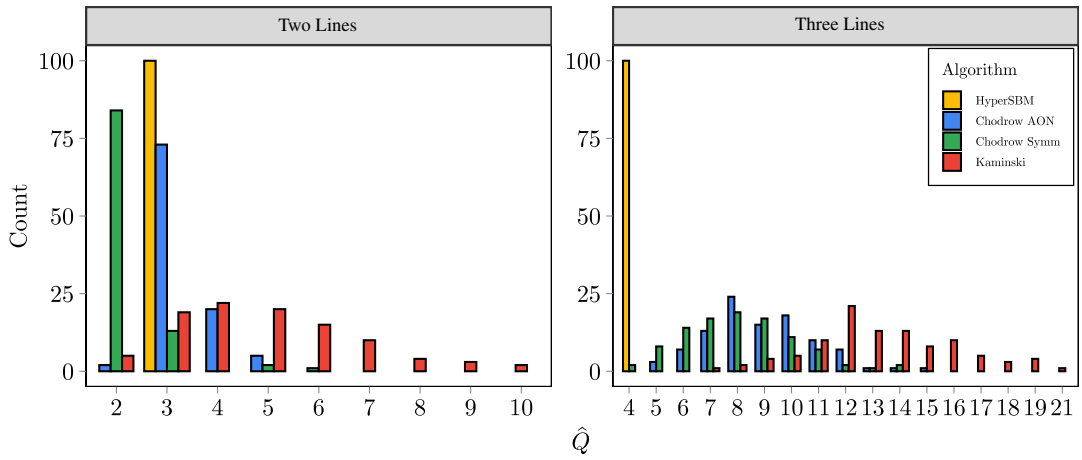

FIGURE 5 Estimated number of groups $\hat{Q}$ on the line clustering problem. Left: two lines (true value of Q is 3), right: three lines (true value of Q is 4).

despite both methods being based on a Stochastic Block Model (SBM) framework with a maximum-likelihood approach.

4 | ANALYSIS OF A CO-AUTHORSHIP DATASET

4.1 | Dataset description

We analyze a co-authorship dataset available at http://vlado.fmf.uni-lj.si/pub/networks/data/2mode/Sandi/Sandi.htm. The dataset originates from the bibliography of the book "Product Graphs: Structure and recognition" by Imrich and Klavzăr and is provided as a bipartite author/article graph. To construct the hypergraph, following the approach of Estrada and Rodríguez-Velázquez (2006), we consider authors as nodes and create hyperedges that link authors who have collaborated on the same paper. Further details regarding the dataset pre-treatment can be found in Section G of the Appendix S1, along with additional analyses. In our analysis, we set $M = 4$ and focused on the main connected component of the hypergraph, which consists of 79 authors and 76 hyperedges. Among these hyperedges, 68.5% have a size of 2, while 29% have a size of 3, and 2.5% have a size of 4.

4.2 | Analysis with `HyperSBM`

We conducted an analysis of this dataset using our `HyperSBM` package. The model selection based on the ICL criterion determined that there are two groups ($\hat{Q} = 2$). One group consists of only eight authors, while the remaining 71 authors belong to the second group. Table 5 displays the distribution of the number of distinct co-authors per author. Within the first group of eight authors, six of them have the highest number of distinct co-authors, while the remaining two authors each have four distinct co-authors.

Coming back to the bipartite graph of authors and (co-authored) papers, we looked at the degree distribution of the authors, given in Table 6. This corresponds to the distribution of the

TABLE 5 Distribution of the number of distinct co-authors per author.

Number of co-authors	1	2	3	4	5	6	7	8	10	11	12
Count	23	27	13	6	2	2	1	1	2	1	1

Note: The first group contains the six authors having the largest number of distinct co-authors (between 7 and 12) plus two authors with four co-authors each.

TABLE 6 Degree distribution of authors in the bipartite graph.

Author degree	1	2	3	4	5	6	7	8	10	13
Count	44	14	6	6	4	1	1	1	1	1

Note: Our first group contains the five most collaborating authors, one of the sixth, plus two authors with degree equal to 4.

number of co-authored papers per author. We observed that five of the eight authors from our first group are the ones that co-published the most, the three others having also high degree (one of degree 5 and two of degree 4). Thus, our first group is made of authors (among) the most collaborative ones, which are also (among) the most prolific ones.

Neither the first nor the second group inferred by `HyperSBM` are communities. Indeed we obtained the following estimated values from the size-2 hyperedges: $\hat{B}_{11} \simeq 4.2\%$ is of the same order as $\hat{B}_{12} \simeq 5.1\%$ while $\hat{B}_{22} \simeq 0.8\%$ is around five times smaller. This means that the first group contains authors that have written with authors from the two groups while the second group is made of authors who have less co-authored papers with people of their own group. Looking now at size-3 hyperedges, we get that $\hat{B}_{111} \simeq 2 \cdot 10^{-4}$; $\hat{B}_{112} \simeq 18 \cdot 10^{-4}$; $\hat{B}_{122} \simeq 7 \cdot 10^{-4}$ and $\hat{B}_{222} \simeq 0.6 \cdot 10^{-4}$. The most important estimated frequency is $\hat{B}_{112}$ that concerns two authors of the small first group co-authoring a paper with one author of the large second group. The second most important estimated frequency is $\hat{B}_{122}$ and is obtained for one author from small first group co-authoring a paper with two authors of the large second group. The remaining frequencies of size-3 hyperedges are negligible. This characterizes further the first groups as being composed by authors that do co-author with their own group as well as with authors from the second one.

Finally, looking now at size-4 hyperedges, the only nonnegligible estimated frequency is obtained for $\hat{B}_{1222} \simeq 4 \cdot 10^{-6}$. We note here that the frequencies $\hat{B}$'s with $m = 3$ or 4 are intrinsically on different scales, as also happens with $m = 2$ or 3. So again, authors from group 1 co-authored with the others authors. (Note that the first group is not large enough for a size-4 $\hat{B}$ frequency with at least two authors in that group 1 to be nonnegligible).

4.3 | Comparison with two other methods

We first compared our approach with the hypergraph spectral clustering (HSC) algorithm proposed in Ghoshdastidar and Dukkipati (2017). Let us recall that spectral clustering does not come with a statistical criterion to select the number of groups. Looking at the partition obtained with $Q = 2$ groups, spectral clustering outputs groups with sizes 24 and 55, respectively. These groups are neither characterized by the number of co-authors nor their degrees in the bipartite graph (see details in Appendix S1). Indeed, in our case the best clusters are not communities and their sizes are very different, while we recall that spectral clustering tends to: (i) extract communities and (ii) favor groups of similar size.

We then analyzed the same dataset as a bipartite graph of authors/papers with the `R` package `sbm` through the function `estimateBipartiteSBM` (Chiquet et al., 2022). This method infers a latent blockmodel (that in fact corresponds to a SBM for bipartite graphs) and automatically selects a number of groups on both parts (authors and papers). The method relies on the same core `VEM` algorithm as ours, adapted to the bipartite graphs context. Hereafter, we refer to this method as the `Bipartite-SBM` implementation. Let us underline here that while the bipartite stochastic blockmodel can be written as a particular case of a HSBM, the converse is not true (see Section A.3 in the Appendix S1). In particular, our hypergraph SBM is not constrained by the need to cluster the set of hyperedges.

The `Bipartite-SBM` also selected two groups of authors (and one group of papers). There was one small group with four authors, entirely contained in our first small group; it corresponds to authors that have the highest degree in the bipartite graph and the highest number of co-authors. So, `Bipartite-SBM` output a very small group of the most prolific and the most collaborative authors in this dataset. Further details about the distinctions between these groups and the ones obtained by `HyperSBM` are given in Appendix S1.

As a conclusion, we see that while the outputs of `Bipartite-SBM` and `HyperSBM` may seem close on this specific dataset, they are nonetheless different. On the other hand, and still on this specific dataset, the spectral clustering approach outputs results that are completely different from those of `HyperSBM`.

5 | DISCUSSION

We have proposed a hypergraph SBM for simple hypergraphs and general clusters types, that is, our work is not limited to community detection and/or equally sized clusters. This is in sharp contrast with most existing approaches. For example, Ghoshdastidar and Dukkipati (2014, 2017) obtained error bounds that converge to zero only for the (**Aff-m**) model with equally sized groups and assuming moreover that $\alpha^{(m)} > \beta^{(m)}$. Moreover, references such as Ke et al. (2020), Ahn et al. (2018), and Chien et al. (2019) primarily focus on community detection, which means they only identify clusters that correspond to communities. Our inference procedure is based on a maximum-likelihood approach, which should in principle provide some statistical guarantees. While consistency and asymptotic normality of the variational and the maximum likelihood estimators in our HSBM is left for future work, we believe that such results could be obtained following approaches used in the context of graphs SBMs (Bickel et al., 2013; Celisse et al., 2012). It is worth noting that while Chodrow et al. (2021) initially employ a maximum likelihood approach, they deviate from that setting for their inference procedure. In contrast, our method retains the maximum likelihood framework throughout the inference process. The maximum likelihood approach also enables the use of a penalized criterion for model selection. The SBM for hypergraphs presented in Balasubramanian (2021) is highly general. However, their least-squares estimator for a hypergraphon model is computationally infeasible. Additionally, their Algorithm 1 is dedicated to community detection and does not provide general cluster recovery.

Our model can accommodate self-loops without significant changes by allowing for $m = 1$. Furthermore, it can be easily extended to handle multiple hypergraphs (with or without self-loops) by incorporating a zero-inflated or deflated Poisson distribution on the conditional distribution of the hyperedges. In a more general setting, the conditional Bernoulli distribution can be replaced with any parametric distribution to handle weighted hypergraphs, and it

could also easily incorporate covariates. This flexibility allows for the adaptation of our model to various types of hypergraph data.

While an important challenge is to reduce the complexity of our approach, some gain could be provided by constraining the parameter set. For instance, Contisciani et al. (2022) consider a Poisson HSBM, where the connectivity parameter is nonzero only between nodes in the same cluster. While this assumption is quite restrictive, it is mitigated by the introduction of overlapping clusters. In the same way, Ruggeri et al. (2023) propose a similar model where the connectivity parameter is the sum of nodes-pairs contributions, resulting in a model that differs from what could be obtained through a clique reduction graph (namely, the graph obtained from hyperedges transformed into cliques). In both cases, these constraints on the parameters considerably reduce the complexity of the inference procedure which is based on a variational-like approach (but does not rely on an ELBO). We believe that similar techniques could be useful in our case and plan to explore that in future works.

ACKNOWLEDGMENTS

Funding was provided by the French National Research Agency (ANR) grant ANR-18-CE02-0010-01 EcoNet. L. Brusa acknowledges the financial support from the grant "Hidden Markov Models for Early Warning Systems" of Ministero dell'Università e della Ricerca (PRIN 2022TZEXKF). Open access publishing facilitated by Universita degli Studi di Milano-Bicocca, as part of the Wiley - CRUI-CARE agreement.

ORCID

Luca Brusa https://orcid.org/0000-0002-8156-470X

REFERENCES

Ahn, K., Lee, K., & Suh, C. (2018). Hypergraph spectral clustering in the weighted stochastic block model. *IEEE Journal of Selected Topics in Signal Processing*, *12*(5), 959–974.

Allman, E., Matias, C., & Rhodes, J. (2011). Parameters identifiability in a class of random graph mixture models. *Journal of Statistical Planning and Inference*, *141*, 1719–1736.

Balasubramanian, K. (2021). Nonparametric modeling of higher-order interactions via hypergraphons. *Journal of Machine Learning Research*, *22*(146), 1–35.

Battiston, F., Cencetti, G., Iacopini, I., Latora, V., Lucas, M., Patania, A., Young, J.-G., & Petri, G. (2020). Networks beyond pairwise interactions: Structure and dynamics. *Physics Reports*, *874*, 1–92.

Behrens, S., Erbes, C., Ferrara, M., Hartke, S. G., Reiniger, B., Spinoza, H., & Tomlinson, C. (2013). New results on degree sequences of uniform hypergraphs. *Electronic Journal of Combinatorics*, *20*(4), p14.

Bick, C., Gross, E., Harrington, H. A., & Schaub, M. T. (2023). What are higher-order networks? *SIAM Review*, *65*(3), 686–731.

Bickel, P., Choi, D., Chang, X., & Zhang, H. (2013). Asymptotic normality of maximum likelihood and its variational approximation for stochastic blockmodels. *Annals of Statistics*, *41*(4), 1922–1943.

Biernacki, C., Celeux, G., & 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.

Cafieri, S., Hansen, P., & Liberti, L. (2010). Loops and multiple edges in modularity maximization of networks. *Physical Review E*, *81*, 046102.

Celisse, A., Daudin, J.-J., & Pierre, L. (2012). Consistency of maximum-likelihood and variational estimators in the stochastic block model. *Electronic Journal of Statistics*, *6*, 1847–1899.

Cerqueira, A., & Leonardi, F. (2020). Estimation of the number of communities in the stochastic block model. *IEEE Transactions on Information Theory*, *66*(10), 6403–6412.

Chelaru, M. I., Eagleman, S., Andrei, A. R., Milton, R., Kharas, N., & Dragoi, V. (2021). High-order correlations explain the collective behavior of cortical populations in executive, but not sensory areas. *Neuron*, *109*(24), 3954–3961.

Chien, I. E., Lin, C.-Y., & Wang, I.-H. (2019). On the minimax misclassification ratio of hypergraph community detection. *IEEE Transactions on Information Theory*, *65*(12), 8095–8118.

Chiquet, J., Donnet, S., großBM team, & Barbillon, P. (2022). *Sbm: Stochastic blockmodels*. R Package Version 0.4.4.

Chodrow, P. S. (2020). Configuration models of random hypergraphs. *Journal of Complex Networks*, *8*(3), cnaa018.

Chodrow, P. S., Veldt, N., & Benson, A. R. (2021). Generative hypergraph clustering: From blockmodels to modularity. *Science Advances*, *7*(28), eabh1303.

Cole, S., & Zhu, Y. (2020). Exact recovery in the hypergraph stochastic block model: A spectral algorithm. *Linear Algebra and its Applications*, *593*, 45–73.

Côme, E., & Latouche, P. (2015). Model selection and clustering in stochastic block models based on the exact integrated complete data likelihood. *Statistical Modelling*, *15*(6), 564–589.

Contisciani, M., Battiston, F., & De Bacco, C. (2022). Inference of hyperedges and overlapping communities in hypergraphs. *Nature Communications*, *13*, 7229.

Daudin, J.-J., Picard, F., & Robin, S. (2008). A mixture model for random graphs. *Statistics and Computing*, *18*(2), 173–183.

Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, *39*, 1–38.

Deng, C., Xu, X.-J., & Ying, S. (2024). Strong consistency of spectral clustering for the sparse degree-corrected hypergraph stochastic block model. *IEEE Transactions on Information Theory*, *70*(3), 1962–1977.

Dumitriu, I., Wang, H., & Zhu, Y. (2022). *Partial recovery and weak consistency in the non-uniform hypergraph stochastic block model* (Technical Report). arXiv:2112.11671.

Estrada, E., & Rodríguez-Velázquez, J. A. (2006). Subgraph centrality and clustering in complex hyper-networks. *Physica A*, *364*, 581–594.

Frank, O., & Harary, F. (1982). Cluster inference by using transitivity indices in empirical graphs. *Journal of the American Statistical Association*, *77*(380), 835–840.

Ghoshal, G., Zlatić, V., Caldarelli, G., & Newman, M. E. J. (2009). Random hypergraphs and their applications. *Physical Review E*, *79*, 066118.

Ghoshdastidar, D., & Dukkipati, A. (2014). *Consistency of spectral partitioning of uniform hypergraphs under planted partition model*. In *Advances in neural information processing systems* (Vol. *27*). NIPS.

Ghoshdastidar, D., & Dukkipati, A. (2017). Consistency of spectral hypergraph partitioning under planted partition model. *The Annals of Statistics*, *45*(1), 289–315.

Holland, P., Laskey, K., & Leinhardt, S. (1983). Stochastic blockmodels: Some first steps. *Social Networks*, *5*, 109–137.

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification*, *2*, 193–218.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, *37*(2), 183–233.

Kamiński, B., Poulin, V., Prałat, P., Szufel, P., & Théberge, F. (2019). Clustering via hypergraph modularity. *PLoS One*, *14*(11), e0224307.

Ke, Z. T., Shi, F., & Xia, D. (2020). *Community detection for hypergraph networks via regularized tensor power iteration* (Technical Report). arXiv:1909.06503.

Kruskal, J. (1977). Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra and its Applications*, *18*(2), 95–138.

Leordeanu, M., & Sminchisescu, C. (2012). *Efficient hypergraph clustering*. In N. D. Lawrence & M. Girolami (Eds.), *Proceedings of the fifteenth international conference on artificial intelligence and statistics Proceedings of machine learning research* (Vol. *22*, pp. 676–684). PMLR.

Massen, C. P., & Doye, J. P. K. (2005). Identifying communities within energy landscapes. *Physical Review E*, *71*, 046101.

Matias, C., & Robin, S. (2014). Modeling heterogeneity in random graphs through latent space models: A selective review. *ESAIM: Proceedings and Surveys*, *47*, 55–74.

Muyinda, N., De Baets, B., & Rao, S. (2020). Non-king elimination, intransitive triad interactions, and species coexistence in ecological competition networks. *Theoretical Ecology*, *13*, 385–397.

Newman, M. E. J. (2016). Equivalence between modularity optimization and maximum likelihood methods for community detection. *Physical Review E*, *94*, 052315.

Ng, T., & Murphy, T. (2022). Model-based clustering for random hypergraphs. *Advances in Data Analysis and Classification*, *16*, 691–723.

Poda, V., & Matias, C. (2024). Comparison of modularity-based approaches for nodes clustering in hypergraphs. *Peer Community Journal*, *4*, e37.

Rohe, K., Chatterjee, S., & Yu, B. (2011). Spectral clustering and the high-dimensional stochastic blockmodel. *Annals of Statistics*, *39*(4), 1878–1915.

Ruggeri, N., Contisciani, M., Battiston, F., & Bacco, C. D. (2023). Community detection in large hypergraphs. *Science Advances*, *9*(28), eadg9159.

Singh, P., & Baruah, G. (2021). Higher order interactions and species coexistence. *Theoretical Ecology*, *14*, 71–83.

Squartini, T., & Garlaschelli, D. (2011). Analytical maximum-likelihood method to detect patterns in real networks. *New Journal of Physics*, *13*(8), 083001.

Stephan, L., & Zhu, Y. (2022). *Sparse random hypergraphs: Non-backtracking spectra and community detection*. In *2022 IEEE 63rd annual symposium on foundations of computer science (FOCS)* (pp. 567–575). IEEE.

Swan, M., & Zhan, J. (2021). Clustering hypergraphs via the MapEquation. *IEEE Access*, *9*, 72377–72386.

Torres, L., Blevins, A. S., Bassett, D., & Eliassi-Rad, T. (2021). The why, how, and when of representations for complex systems. *SIAM Review*, *63*(3), 435–485.

Turnbull, K., Lunagómez, S., Nemeth, C., & Airoldi, E. (2023). Latent space modeling of hypergraph data. *Journal of the American Statistical Association*, 1–13. https://doi.org/10.1080/01621459.2023.2270750

van Handel, R. (2011). On the minimal penalty for Markov order estimation. *Probability Theory and Related Fields*, *150*(3), 709–738.

Vazquez, A. (2009). Finding hypergraph communities: A Bayesian approach and variational solution. *Journal of Statistical Mechanics: Theory and Experiment*, *2009*(7), P07006.

Wolff, K. H. (1950). *The sociology of Georg Simmel*. The Free Press.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: Brusa, L., & Matias, C. (2024). Model-based clustering in simple hypergraphs through a stochastic blockmodel. *Scandinavian Journal of Statistics*, 1–24. https://doi.org/10.1111/sjos.12754