A stochastic block model for hypergraphs

Luca Brusa[1] and Catherine Matias[2,3,4]

[1]University of Milano-Bicocca, Milano, Italy
[2] Centre National de la Recherche Scientifique, Paris, France
[3] Sorbonne Université, Paris, France
[4] Université de Paris Cité, Paris, France

Warwick - Sept 2023

Outline

1. The need for higher-order interactions
2. Stochastic blockmodel for hypergraphs
3. Experiments
4. Conclusions and perspectives

Higher-order interactions I

Motivations

- Networks or graphs focus on pairwise interactions
- These type of pairwise interactions can already be quite elaborate: undirected/directed, binary/weighted, simple/multiple, static/dynamic, multiplex or multi-layers, . . .
- Nonetheless pairwise interactions are not sufficient to describe the nature of complex interactions :
  - e.g. the presence of a 3rd chemical component may modify the interaction of 2 other ;
- Collective interactions or group interactions are richer than just pairwise interactions

$\hookrightarrow$ These are called higher-order interactions (HOI).

Higher-order interactions II

Where do we find HOI?

- Social networks: triadic and larger groups (as early as Simmel, 1950)
- Scientific co-authorship,
- Interactions between chemical components,
- Interactions between neurons in brain networks,
- etc

These interactions **CAN NOT** be represented by a graph.

Higher-order interactions III

This is a nice recent review (2020):

Contents lists available at ScienceDirect

Physics Reports

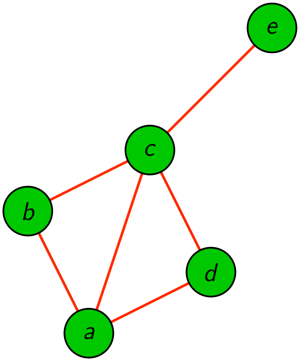
journal homepage: www.elsevier.com/locate/physrep

Networks beyond pairwise interactions: Structure and dynamics

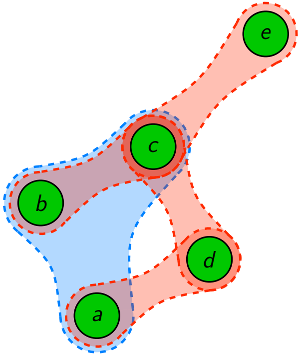
Federico Battiston [a,*], Giulia Cencetti [b], Iacopo Iacopini [c,d], Vito Latora [c,e,f,g], Maxime Lucas [h,i,j], Alice Patania [k], Jean-Gabriel Young [l], Giovanni Petri [m,n]

Pairwise vs HOI

HOI are defined as sets of interacting entities.
e.g. $V = \{a, b, c, d, e\}; \mathcal{I} = \{\{a, b, c\}, \{a, d\}, \{c, d\}, \{c, e\}\}$

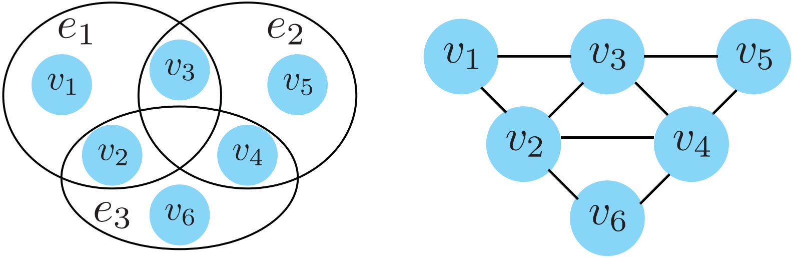


(a) Pairwise interactions



(b) A HOI in blue

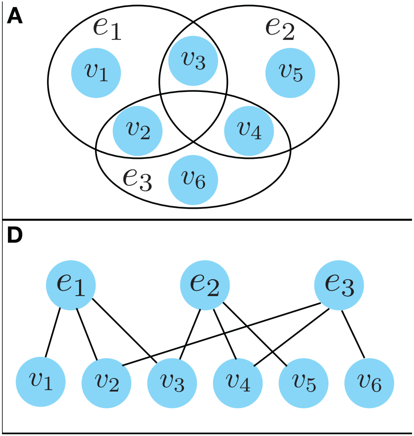
Naïve Graph representation: clique expansion graph



Picture from Schaub *et al.* 2021

- Each interaction is transformed into a clique = all edges between pairs are present ;
- HOIs actually disappeared !
- Too simplistic: For e.g, in co-authorship 1 paper with 3 authors $\neq$ 3 different papers written by pairs of those authors.

Bipartite graph representation (two-modes network or star-expansion graph)



Picture from Schaub *et al.* 2021

- No loss of information;
- But "higher-order" now translates into node degrees in one part;
- 2 two parts don't play symmetric roles: statistical models on bipartite graphs are not appropriate here

Simple hypergraphs

Definition

A (simple) hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ is defined as a set of nodes $\mathcal{V} \neq \emptyset$ and a set of hyperedges $\mathcal{E}$. **Each hyperedge is a non-empty collection of m distinct nodes** $(2 \leq m \leq M)$ taking part within an interaction.

- Hypergraphs naturally include the entity of graphs, by simply considering hyperedges of size $m = 2$;
- A hypergraph can contain a size-3 hyperedge $[a, b, c]$ without any requirement on the existence of the size-2 hyperedges $[a, b]$, $[a, c]$, and $[b, c]$.

Clustering the nodes of a hypergraph I

What has been done up to now

- **Modularity-based** approaches
  - Different hypergraph modularity definitions: what kind of communities do they favour?
  - Note that for computational reasons, these focus on ***multisets-hypergraphs*** where nodes may be repeated in a same hyperedge;
  - This is not always appropriate, e.g. co-authorship dataset;
  - In the context of graphs, absence of self-loops and multiple edges are known to generate pbms in modularity approaches
- **Spectral clustering** has been generalized to hypergraphs but
  - it tends to favour groups of the same size;
- **Challenges**
  - Look for general clusters and not only *communities*
  - None of these methods comes with a statistical criterion to select the number of groups Q

Clustering the nodes of a hypergraph II

Our proposal

- We focus on **simple** hypergraphs (instead of multisets-hypergraphs);
- We define **a stochastic blockmodel** to cluster the nodes of a hypergraph
  - We establish **parameter identifiability** results;
  - We propose a **variational expectation-maximisation** algorithm to infer clusters and parameters;
  - We propose an **ICL criterion** to select the number of clusters;
  - All these tools are implemented (in `C++`) in a efficient `R` **package** called `HyperSBM`.

Outline

SBM formulation

- $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, with $\mathcal{V} = \{1, \ldots, n\}$ nodes and $\mathcal{E}$ hyperedges;
- For each $2 \leq m \leq M$, let $\mathcal{V}^{(m)} = \big\{\{i_1, \ldots, i_m\} : i_1, \ldots, i_m \in \mathcal{V} \text{ and } i_1 \neq \ldots \neq i_m\big\}$, set of unordered node tuples of size m;
- **Observations:** At each $\{i_1, \ldots, i_m\} \in \mathcal{V}^{(m)}$, we observe indicator variable $Y_{i_1,\ldots,i_m} = 1\{\{i_1, \ldots, i_m\} \in \mathcal{E}\}$;
- **Latent clusters:** $Z_1, \ldots, Z_n$ iid in $\{1, \ldots, Q\}$ with $\pi_q = \mathbb{P}(Z_i = q)$;
- **Conditional independence assumption**:
$\{Y_{i_1,\ldots,i_m}\}_{\{i_1,\ldots,i_m\}\in\mathcal{V}^{(m)}} | \{Z_1, \ldots, Z_n\}$ are independent with
$Y_{i_1,\ldots,i_m} | \{Z_1 = q_1, \ldots, Z_m = q_m\} \sim \text{Bern}(B^{(m)}_{q_{i_1},\ldots,q_{i_m}})$.

Parameter (generic) identifiability

Generic identifiability: a parameter θ almost surely (*w.r.t.* Lebesgue measure) uniquely defines the distribution $\mathbb{P}_\theta$ (up to label switching on the node groups).

Theorem

For any Q, the parameter $\theta = (\pi_q, B^{(m)}_{q_1,\dots,q_m})_{m,q,q_1,\dots,q_m}$ of the HSBM for (simple) hypergraphs over n nodes, is generically identifiable for large enough n.

Said differently, there is a finite set $\mathcal{C}$ of (non explicit) polynomial conditions on θ such that whenever $\theta \notin \mathcal{C}$, the distribution $\mathbb{P}_\theta$ is uniquely defined by θ.

Inference through variational EM I

- Direct computation of the likelihood is not feasible for large n;
- EM algorithm neither feasible because latent variables are not independent conditional on observed ones;
- Variational approximation to EM algorithm: replace the intractable posterior distribution by the best approximation (w.r.t. Kullback-Leibler divergence) in a class of simpler (factorised) distributions:

$$\mathbb{Q}_\tau(Z_1, \dots, Z_n) = \prod_{i=1}^{n} \mathbb{Q}_\tau(Z_i) = \prod_{i=1}^{n} \prod_{q=1}^{Q} \tau_{iq}^{Z_{iq}},$$

with the variational parameter $\tau_{iq} = \mathbb{Q}_\tau(Z_i = q) \in [0, 1]$ and $\sum_{q=1}^{Q} \tau_{iq} = 1$, for any $i = 1, \dots, n$ and $q = 1, \dots, Q$.

Inference through variational EM II

Evidence lower bound (ELBO)

$$\begin{aligned} \mathcal{J}(\theta,\tau) &= \mathbb{E}_{\mathbb{Q}_\tau}[\log \mathbb{P}_\theta(\boldsymbol{Y},\boldsymbol{Z})] - \mathbb{E}_{\mathbb{Q}_\tau}[\log \mathbb{Q}_\tau(\boldsymbol{Z})] \\ &= \log \mathbb{P}_\theta(\boldsymbol{Y}) - \mathrm{KL}(\mathbb{Q}_\tau(\boldsymbol{Z}) || \mathbb{P}_\theta(\boldsymbol{Z}|\boldsymbol{Y})) \\ &\leq \log \mathbb{P}_\theta(\boldsymbol{Y}), \end{aligned}$$

with equality iff $\mathbb{Q}_\tau(\boldsymbol{Z})$ is the true posterior $\mathbb{P}_\theta(\boldsymbol{Z}|\boldsymbol{Y})$.

VEM maximises the lower bound $\mathcal{J}(\theta,\tau)$ (with respect to τ and θ) instead of the intractable log-likelihood $\log \mathbb{P}_\theta(\boldsymbol{Y})$

VEM algorithm

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

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

  This is equivalent to minimising the Kullback-Leibler divergence.
  In practice this step is obtained by a fixed-point algorithm.

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

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

  thus updating the value of the model parameters π_q and $B^{(m)}_{q_1,\dots,q_m}$.

Model selection and generalizations

Integrated classification likelihood (ICL)

We select $\hat{q} = \arg\max_q ICL(q)$ where

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

Generalizations

- We have not considered self-loops ($m = 1$) but it's easy to do;
- Binary hyperedge variables could be replaced by counting hyperedges variables, replacing the Bernoulli distribution with, for e.g. (zero-inflated or deflated) Poisson law.

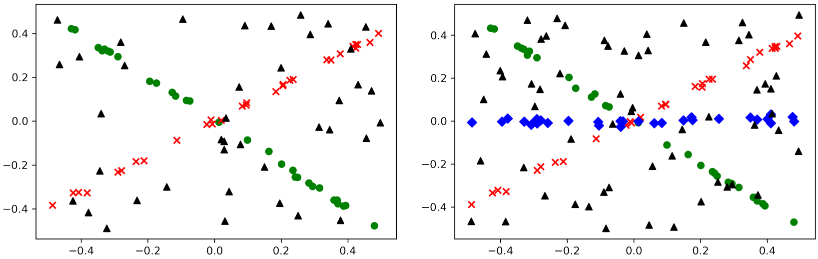
Computational complexity - and considerations over the choice of M

- Focusing on *simple* hypergraphs **has a high price**: we need to explore all the $\binom{n}{m}$ tuples of nodes for all $2 \leq m \leq M$;
- Our algorithm has a complexity of $O(n\binom{n}{M}Q^M)$, which is large;
- Current modularity approaches avoid this issue by working with multisets-hypergraphs, because there the summations over multisets of nodes $\sum_{i_1,\dots,i_m}$ factorize into m independent sums (no constraint that the nodes be different), and this further simplifies the expression of the modularity;
- Again, this is inappropriate on some datasets;
- As a consequence: we recommend to use **a reasonable value of M**: indeed M is not necessarily the largest observed hyperedge size (e.g. co-authorship dataset);

Outline

1. The need for higher-order interactions
2. Stochastic blockmodel for hypergraphs
3. Experiments
4. Conclusions and perspectives

Line clustering through hypergraphs I



Hypergraph construction

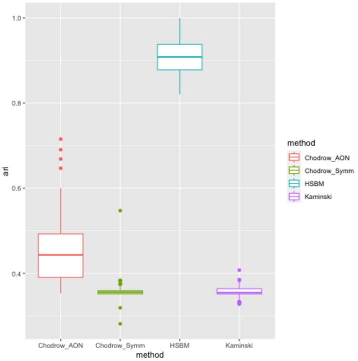
- Select 3 points at random and fit a line
- If residual distance less than a threshold, draw a hyperedge between those 3 points
- Globally set signal:noise hyperedge ratio = 2
- Repeat to obtain 100 3-uniform hypergraphs

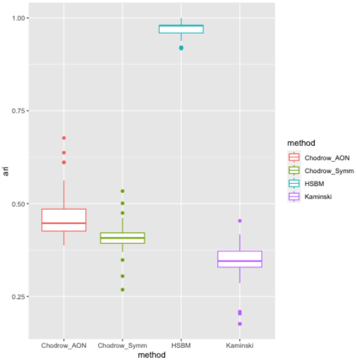
Line clustering through hypergraphs II

Data characteristics

	Pts/line	Noisy pts	Total nb pts	mean nb of hyperedges
2 lines	30	40	100	1070.84
3 lines	30	60	150	587.7

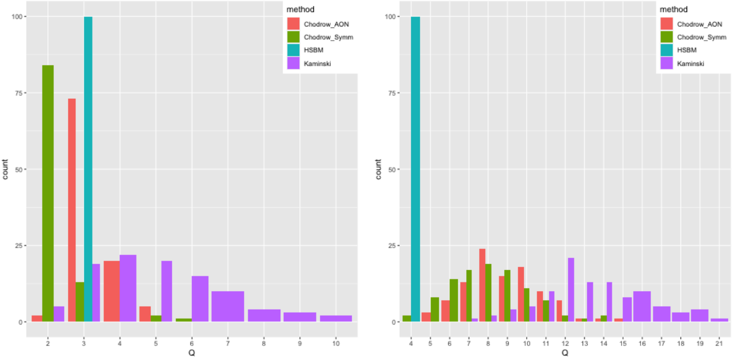
Comparison with modularity based methods I





Adjusted Rand Index

Comparison with modularity based methods II



Estimated number of groups

Outline

1. The need for higher-order interactions
2. Stochastic blockmodel for hypergraphs
3. Experiments
4. Conclusions and perspectives

Conclusions

- We propose a Stochastic Blockmodel for clustering the nodes of a (simple) hypergraph
- We establish (generic) identifiability of the parameters of the model
- Estimation and nodes clustering is performed through VEM algorithm
- ICL criterion is used to select the number of groups
- C++ code wrapped in a R package HyperSBM (https://github.com/LB1304/HyperSBM) and preprint on ArXiV https://arxiv.org/abs/2210.05983

Remaining challenges

- understand the detectability limits for non-uniform hypergraphs ;
- computational issues: explore sparse hypergraphs modelings

Post-doc position on modelling sparse hypergraphs in Paris - deadline for application October, 15th.

Any questions ?

Non equivalence between simple binary hypergraphs and bipartite graphs