An introduction to graph-tool and the minimum description length principle

Complexity 72 Hours Workshop Tutorial

Erik Weis

Complexity 72h

June 24, 2025

Complexity Society Lab

Northeastern University

NSF

Network Science Institute at Northeastern University

Graph-tool

Powerful python package for doing statistical analysis of networks

- Fast computation for scalable analysis of large networks
- **Statistical inference techniques**

The minimum description length (MDL) principle

Scalable statistical framework grounded in Bayesian inference

- Nonparametric models allow for learning arbitrary patterns from data with minimal assumptions (i.e. very large model classes)
- Automatic model selection to avoid *underfitting* and *overfitting* to data
- Analyses with clear and intentional inductive biases

Outline

Part I: Introduction to graph-tool

- Assembling and manipulating graphs
- Overview of package
- Brief performance comparison (shortest paths, clustering)

Part II: Introduction to the minimum description length principle and graph-tool algorithms

Three statistical analyses where MDL is useful

1. Community detection
2. Clustering in bipartite networks
3. Learning networks (and their dynamics) from time series

See notebook: `gt-basics.ipynb`

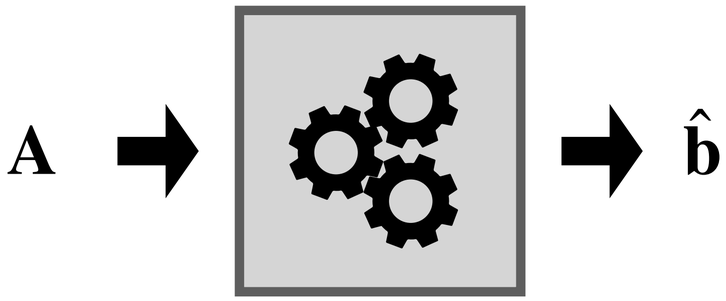
Community detection + MDL + Bayesian inference

Community detection

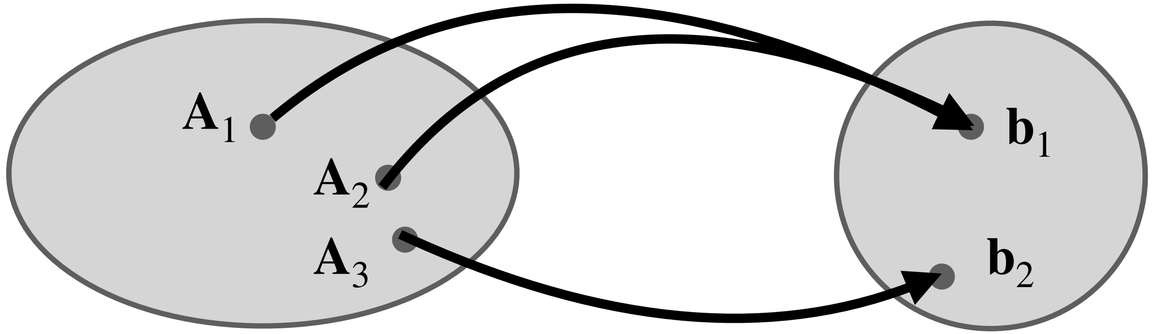
We have a network $\mathbf{A}$ and we want a clustering of the nodes $\hat{\mathbf{b}}$.

A community detection method is either:

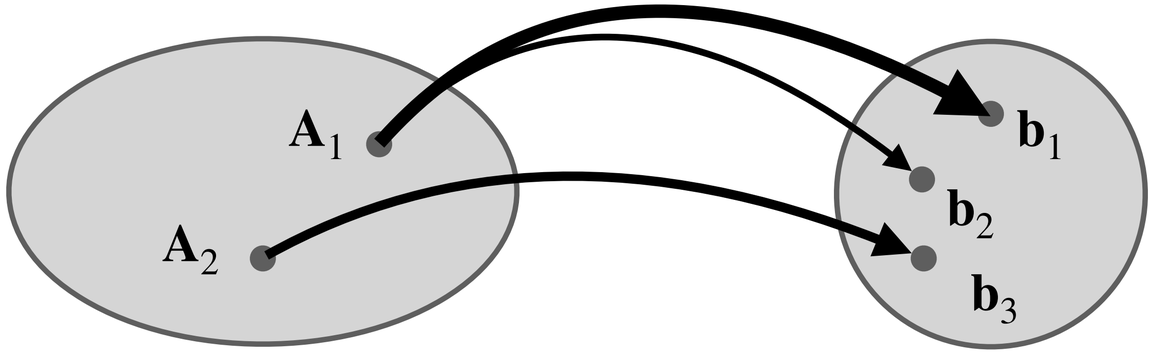
1. a deterministic function $\mathbf{b} = f(\mathbf{A})$
2. a probabilistic function $\mathbf{b} \sim P(\cdot \,|\, \mathbf{A})$



Deterministic methods



Probabilistic methods



This probabilistic mapping establishes a joint probability distribution $P(\mathbf{A}, \mathbf{b})$

A note on generative modeling

Every method is *implicitly* a generative model

All methods define the relationship $P(\mathbf{A}\,|\,\mathbf{b})$, either implicitly or explicitly.

Explicit probabilistic models are advantageous because they have clear and *interpretable* inductive biases.

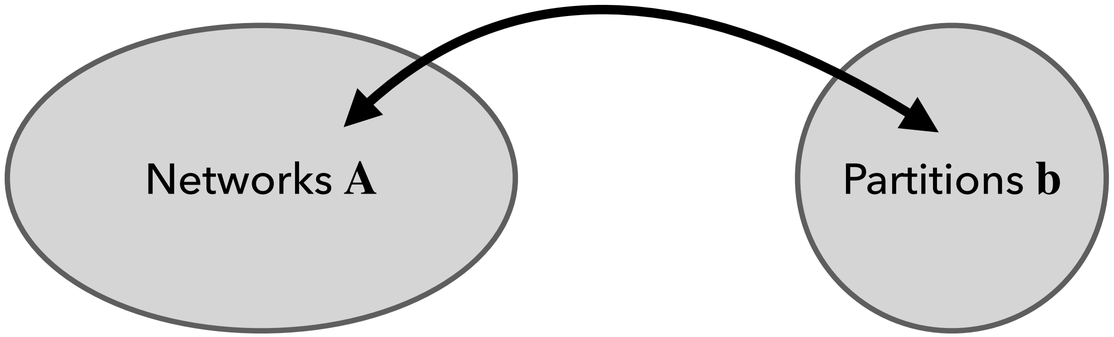
$$P(\mathbf{b}\,|\,\mathbf{A}) = \frac{P(\mathbf{A}\,|\,\mathbf{b})P(\mathbf{b})}{P(\mathbf{A})}$$

PHYSICAL REVIEW E **108**, 024309 (2023)

Implicit models, latent compression, intrinsic biases, and cheap lunches in community detection

Tiago P. Peixoto *
Department of Network and Data Science, Central European University, 1100 Vienna, Austria

Alec Kirkley †
Institute of Data Science, University of Hong Kong, Hong Kong;
Department of Urban Planning and Design, University of Hong Kong, Hong Kong;
and Urban Systems Institute, University of Hong Kong, Hong Kong



A note on generative modeling

Every method is *implicitly* a generative model

All methods define the relationship $P(\mathbf{A} \mid \mathbf{b})$, either implicitly or explicitly.

Explicit probabilistic models are advantageous because they have clear and *interpretable* inductive biases.

$$P(\mathbf{b} \mid \mathbf{A}) = \frac{P(\mathbf{A} \mid \mathbf{b})P(\mathbf{b})}{P(\mathbf{A})}$$

PHYSICAL REVIEW E **94**, 052315 (2016)

"…which in effect means that *a priori* the sizes of all groups are the same and hence that modularity maximization implicitly prefers groups of uniform size, which could also hurt performance if this assumption doesn't match the properties of the observed network."

Departm *USA*

resolution parameter.

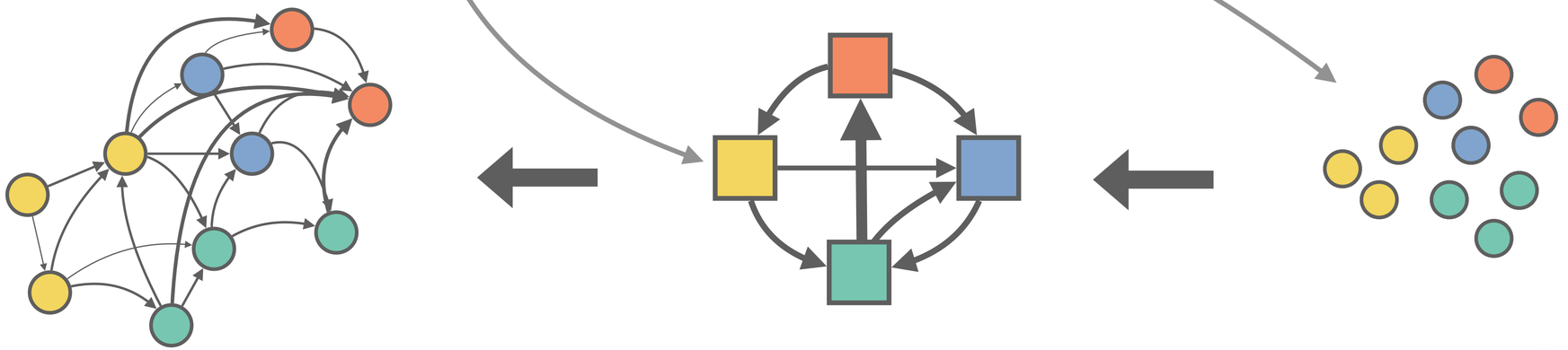
Basic overview of the microcanonical SBM

A generative network model for community structure

The SBM is defined as

$$P(\mathbf{A}, \mathbf{b}) = \underbrace{P(\mathbf{A} \mid \mathbf{e}, \mathbf{b})}_{\text{Likelihood}} \underbrace{P(\mathbf{e} \mid \mathbf{b}) P(\mathbf{b})}_{\text{Priors}}$$

$P(X \mid Y)$: the probability distribution of random variable X *given* we know Y

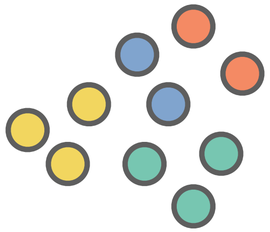


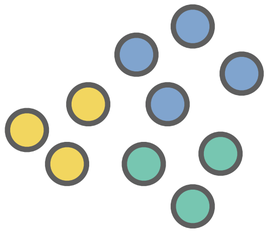
Graph $\mathbf{A}$ | Block graph $\mathbf{e}$ | Partition $\mathbf{b}$

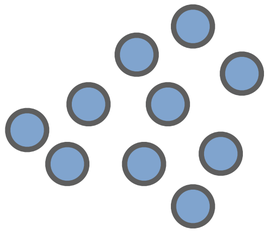
Basic overview of the SBM

Many ways to generate the same data

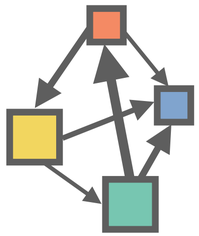
Partition

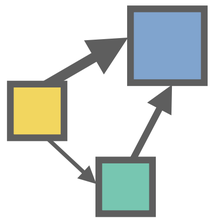


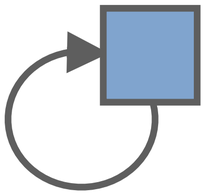




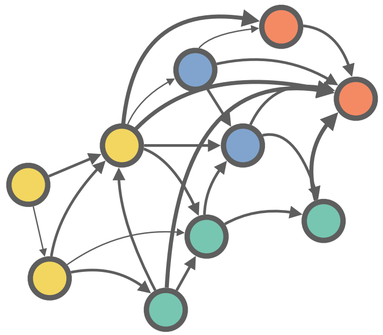
Block graph

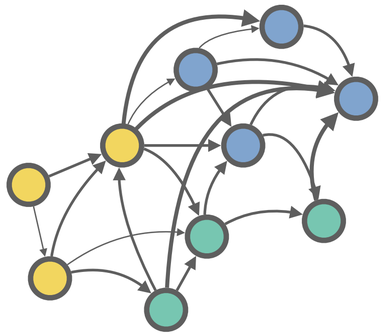


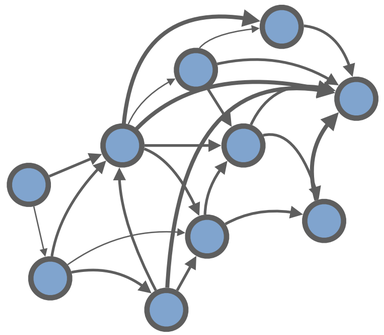




Graph







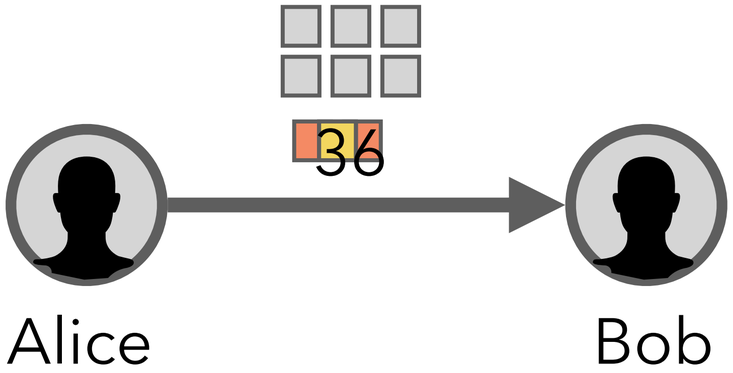
The minimum description length (MDL) principle

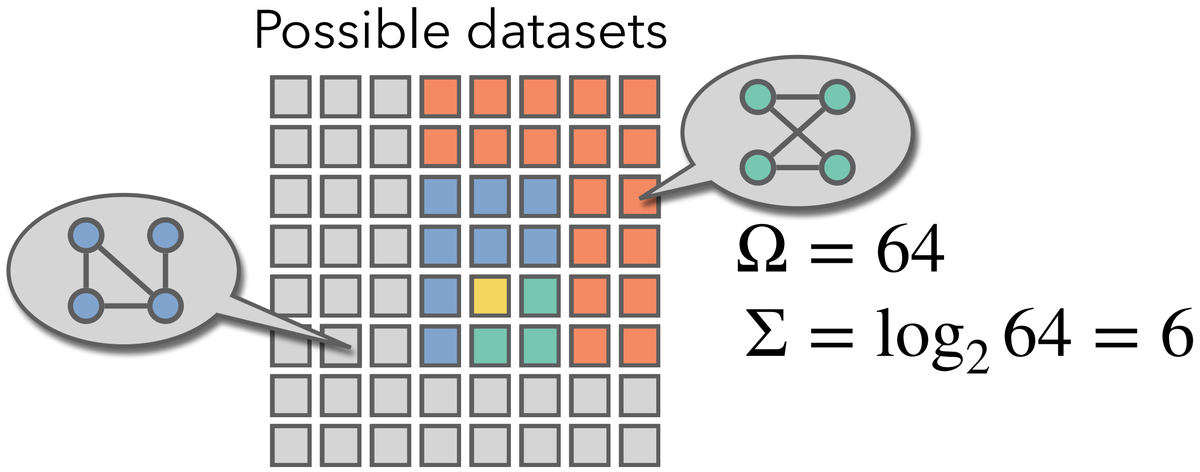
Imagine we need to transmit data X across a communication channel

↪use an encoding scheme.

We can also use a multi-step encoding scheme, allowing us to transmit parts chunks of the data separately.

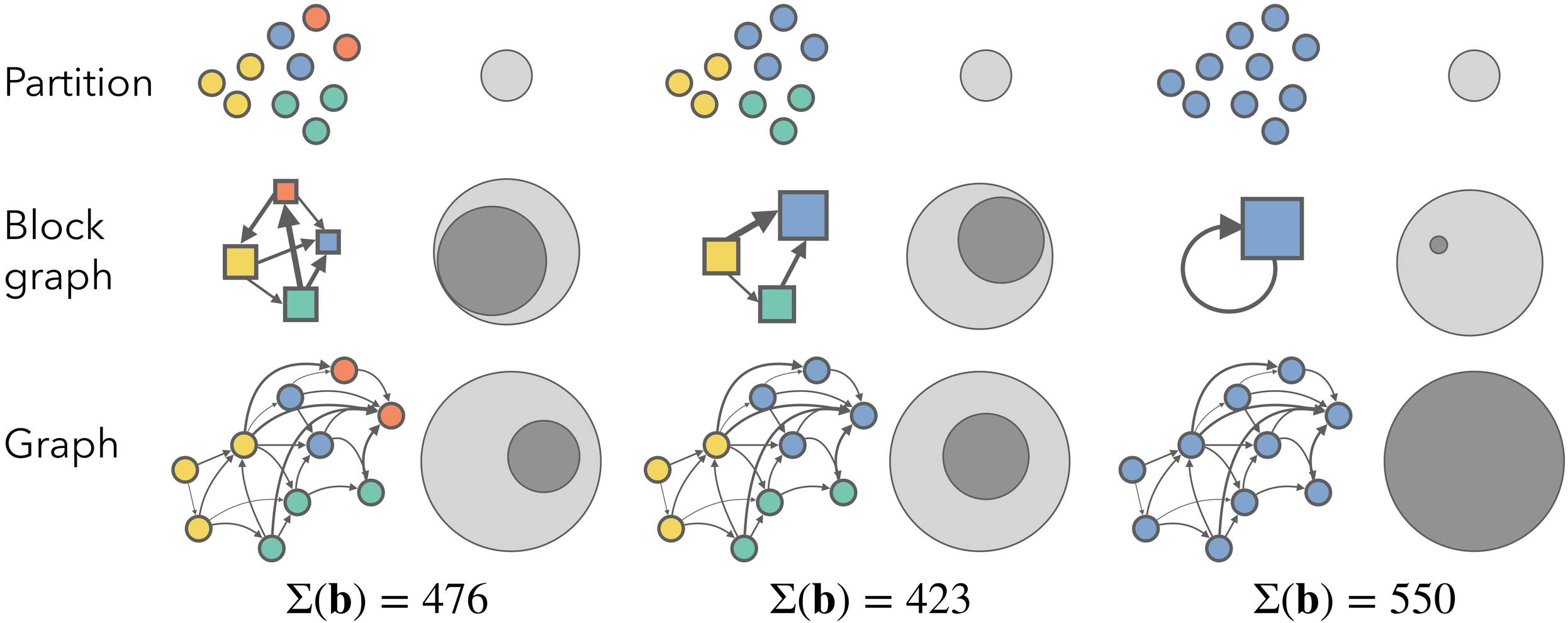
Allows for data *compression*, meaning fewer overall bits required





Community detection using MDL

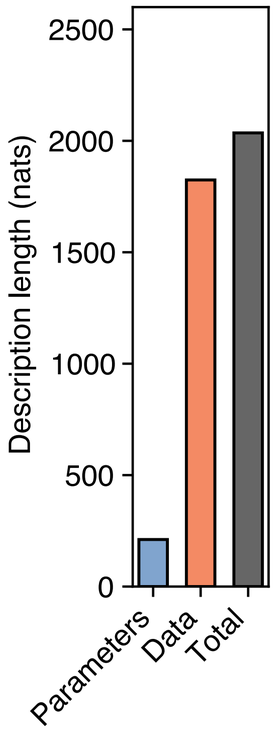
Balancing model complexity with model fit

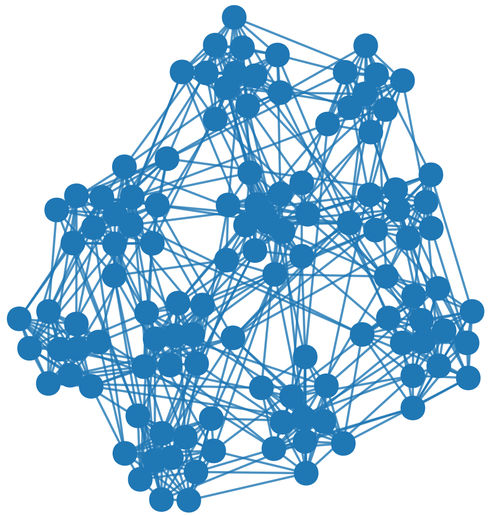


Community detection using MDL

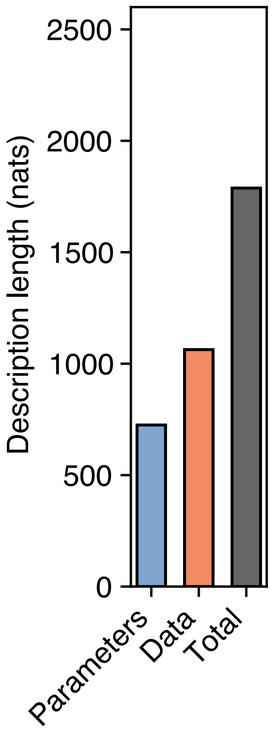
Balancing model complexity with model fit

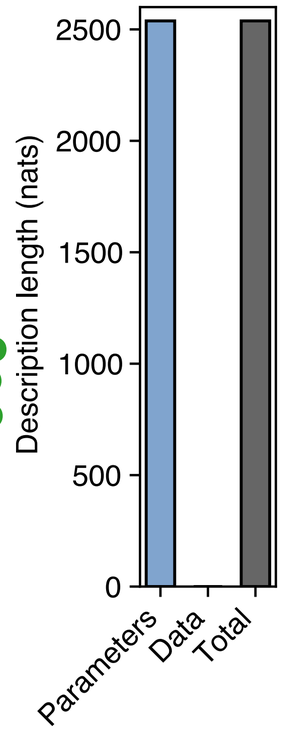
Underfit



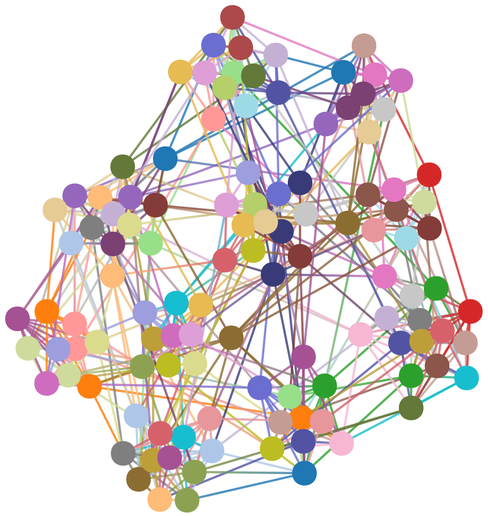


Optimal



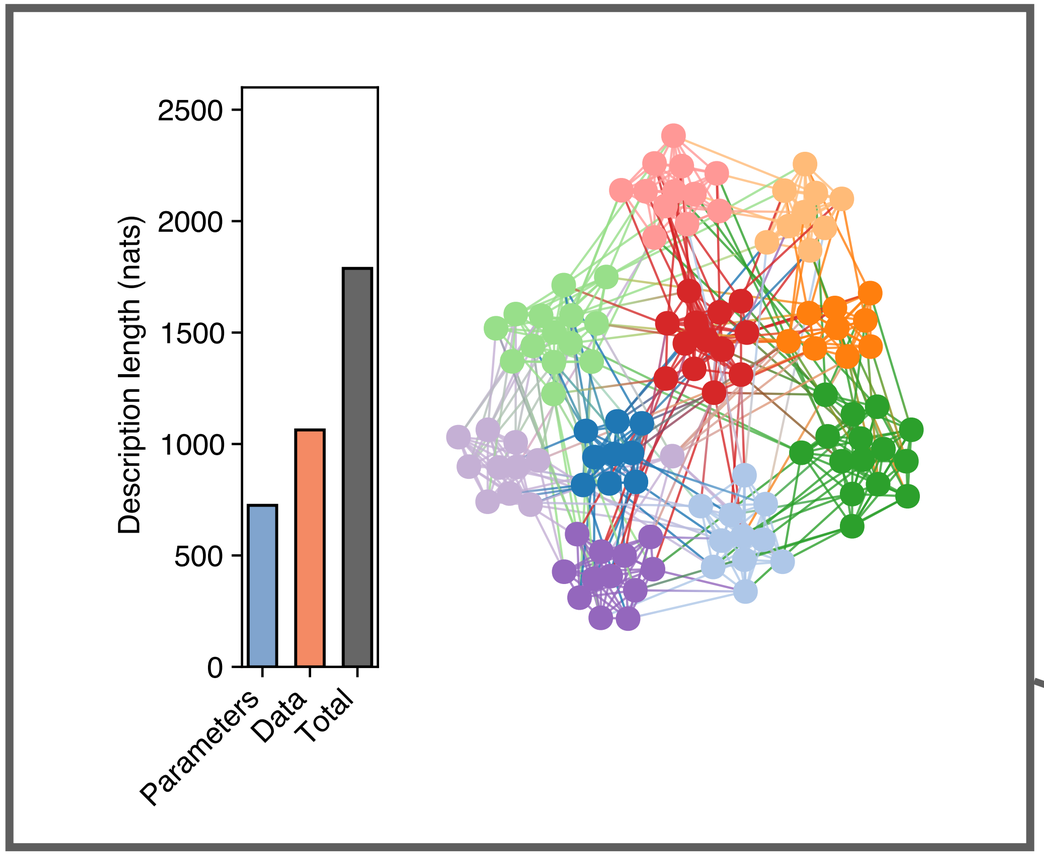


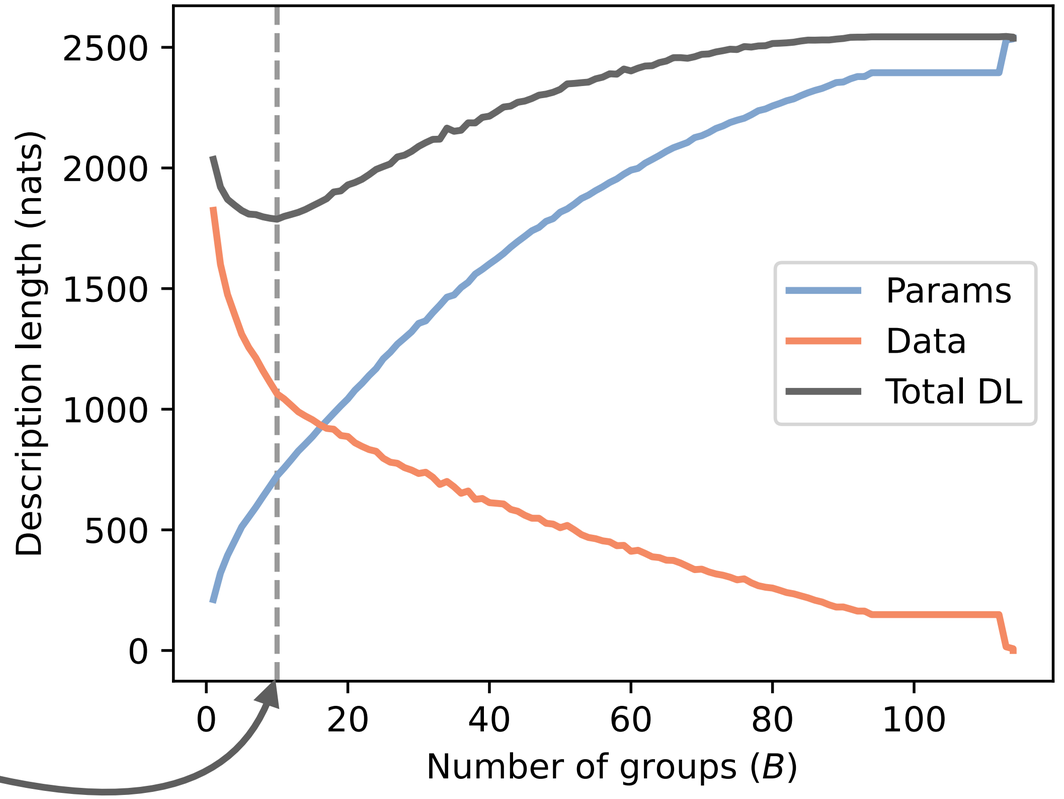
Overfit



Why MDL?

Model selection and Occam's razor



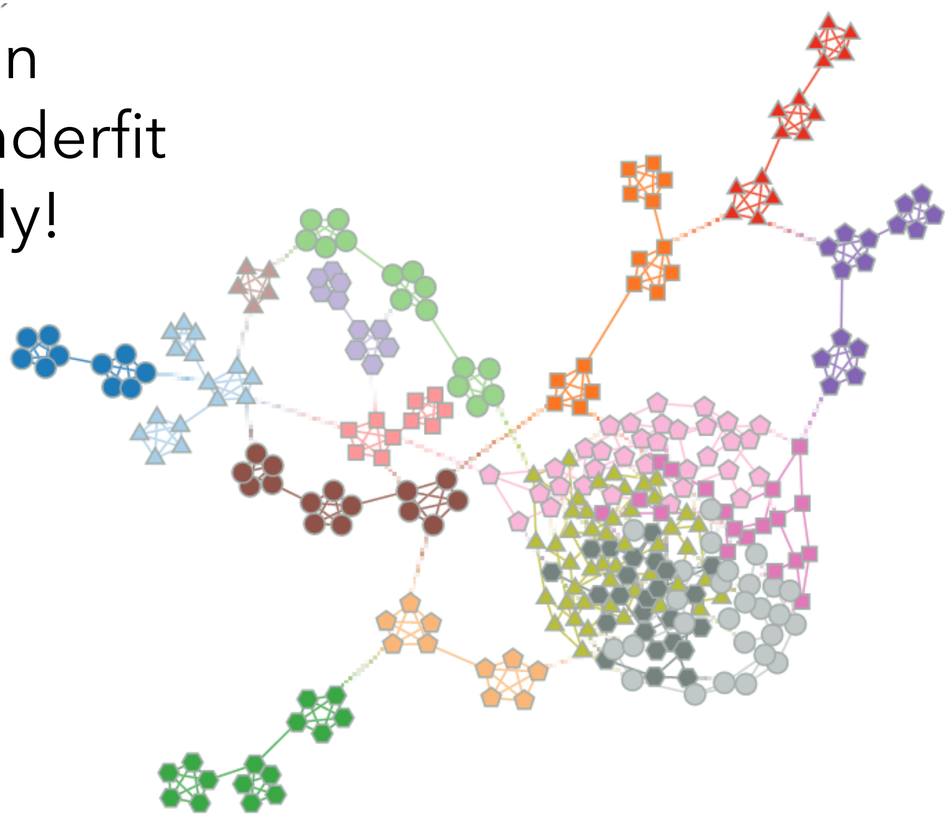


Low complexity ⟶ High complexity

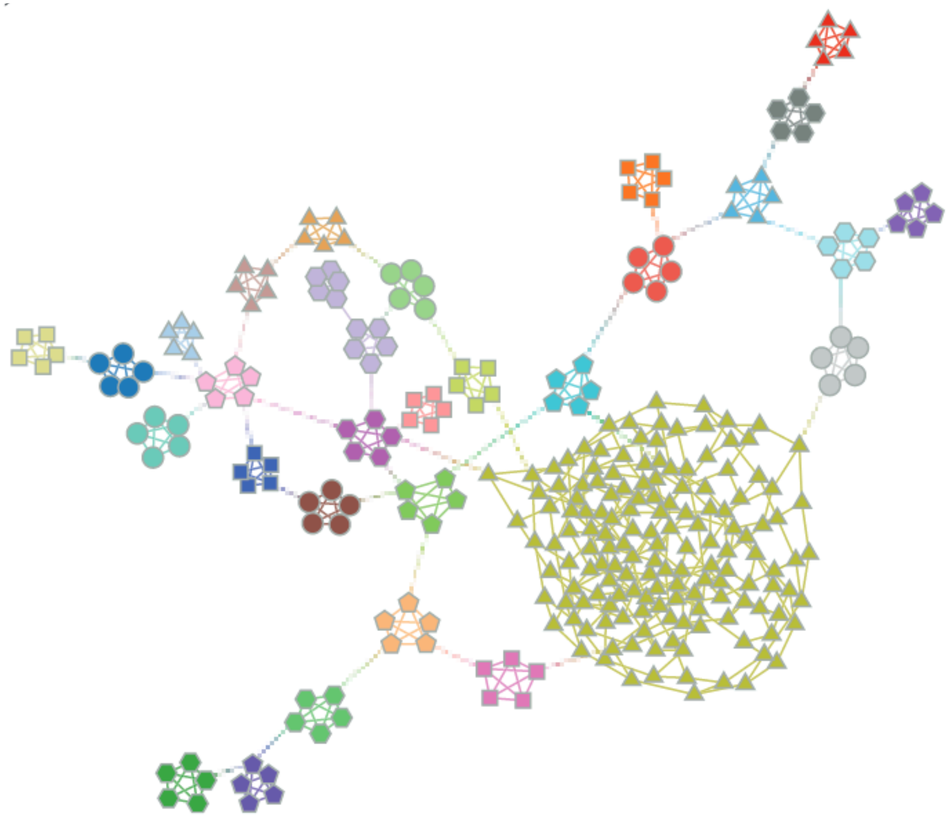
Why MDL?

Model selection and Occam's razor

Modularity can overfit and underfit simultaneously!



Modularity maximization

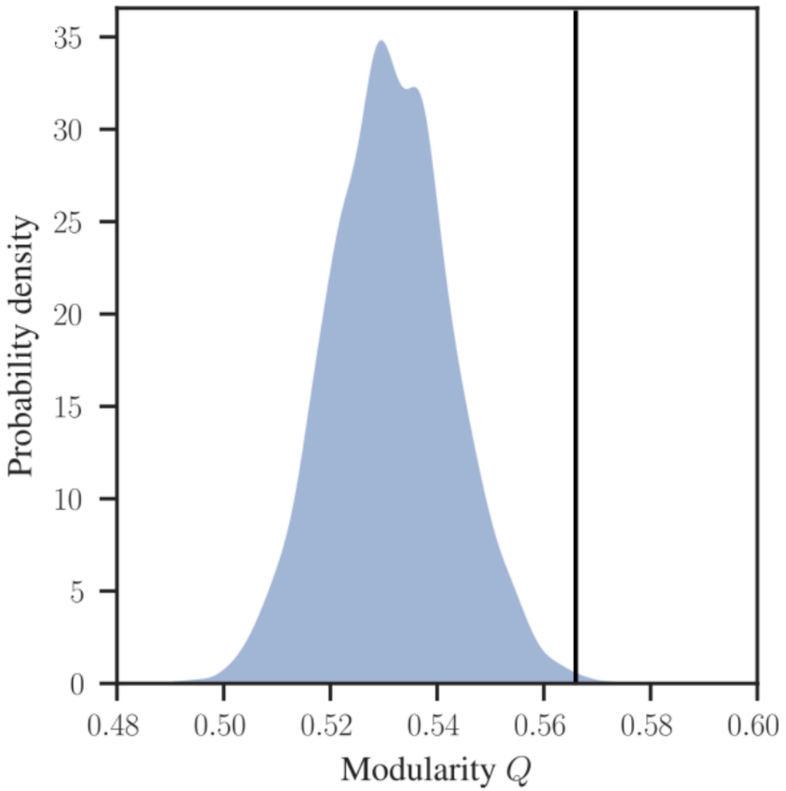


SBM inference

Peixoto. "Descriptive vs. inferential community detection in networks." *Cambridge University Press (2023)*

Why MDL?

Model selection and Occam's razor



Modularity

SBM Inference

Peixoto. "Significant community structure via statistical tests?" (*https://skewed.de/lab/posts/significant-modularity/*)

Why MDL?

Connections to Bayesian inference

The partition that minimizes the description length maximizes the posterior, i.e.

$$\hat{\mathbf{b}} = \underset{\mathbf{b}}{\operatorname{argmin}}\ \Sigma(\mathbf{b})$$
$$= \underset{\mathbf{b}}{\operatorname{argmin}}\ P(\mathbf{b}\,|\,\mathbf{A})$$

*Note: Bayesian inference is more general, as we can sample from the posterior using MCMC

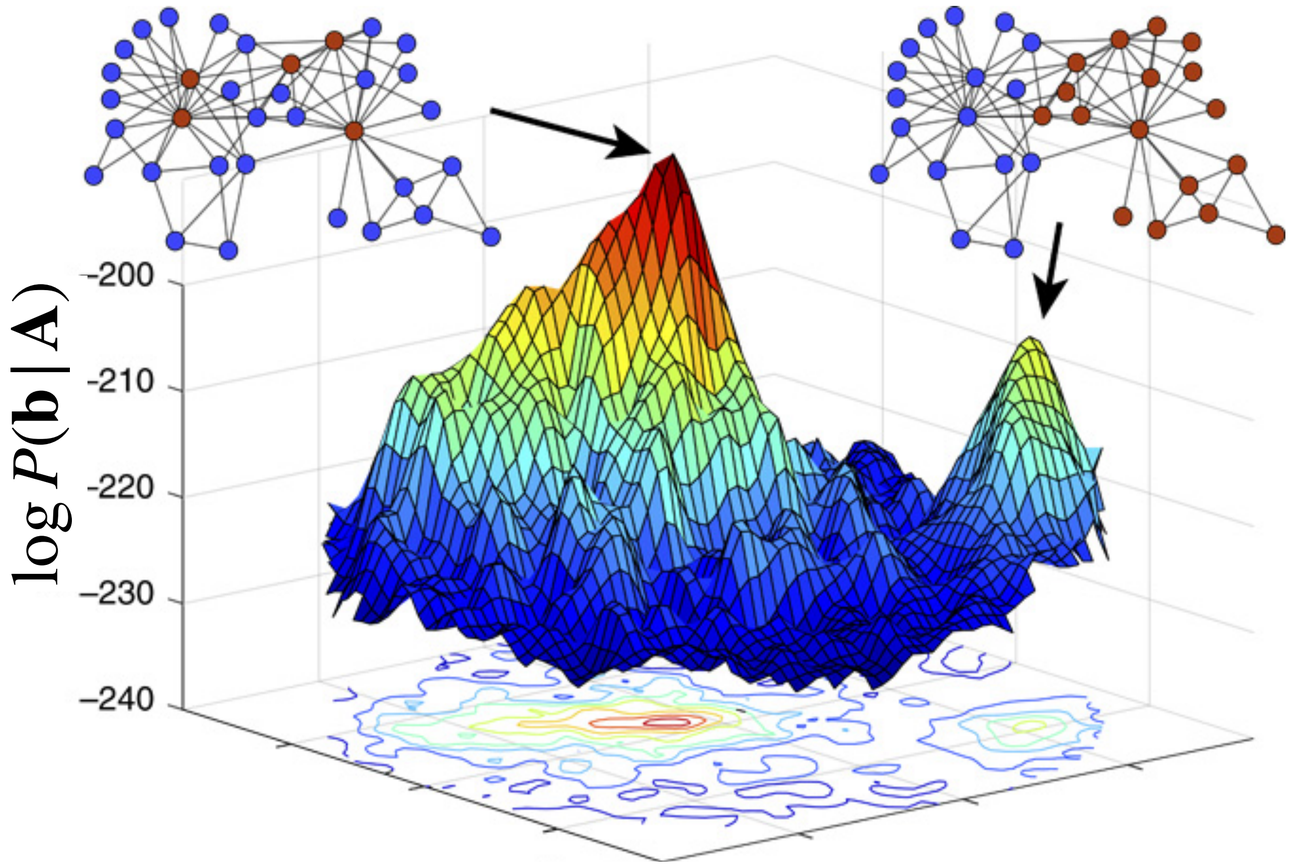


Figure adapted and reinterpreted from Peel et al. "The ground truth about metadata and community detection in networks," *Science Advances (2016)*

MDL and generative models

Two sides of the same coin

The data is a network $\mathbf{A}$, and we assume the receiver knows the number of nodes N and edges E.

1. Transmit the network's degree sequence $\mathbf{k}$.

2. Transmit $\mathbf{A}$, given the degree sequence

MDL encoding		Generative model	
Parameters	Transmit an integer between 0 and $\Omega(\mathbf{k})$	*Prior*	$P(\mathbf{k}) = \frac{1}{\Omega(\mathbf{k})}$
Data	Transmit an integer between 0 and $\Omega(\mathbf{A} \mid \mathbf{k})$	*Likelihood*	$P(\mathbf{A} \mid \mathbf{k}) = \frac{1}{\Omega(\mathbf{A} \mid \mathbf{k})}$

$\Omega(\mathbf{k})$: the number of graphical degree sequences in a network of size N.

$\Omega(\mathbf{A} \mid \mathbf{k})$: the number of networks with the degree sequence $\mathbf{k}$.

MDL and generative models

Two sides of the same coin

	MDL encoding	Generative model
Partition	Transmit an integer between 0 and $\Omega(\mathbf{b})$	$P(\mathbf{b}) = \frac{1}{\Omega(\mathbf{b})}$
Block count matrix	Transmit an integer between 0 and $\Omega(\mathbf{e}\,\vert\,\mathbf{b})$	$P(\mathbf{e}\,\vert\,\mathbf{b}) = \frac{1}{\Omega(\mathbf{e}\,\vert\,\mathbf{b})}$
Graph	Transmit an integer between 0 and $\Omega(\mathbf{A}\,\vert\,\mathbf{e},\mathbf{b})$	$P(\mathbf{A}\,\vert\,\mathbf{e},\mathbf{b}) = \frac{1}{\Omega(\mathbf{A}\,\vert\,\mathbf{e},\mathbf{b})}$

$\Omega(\mathbf{b})$: the number of valid partitions N objects into any number of groups.

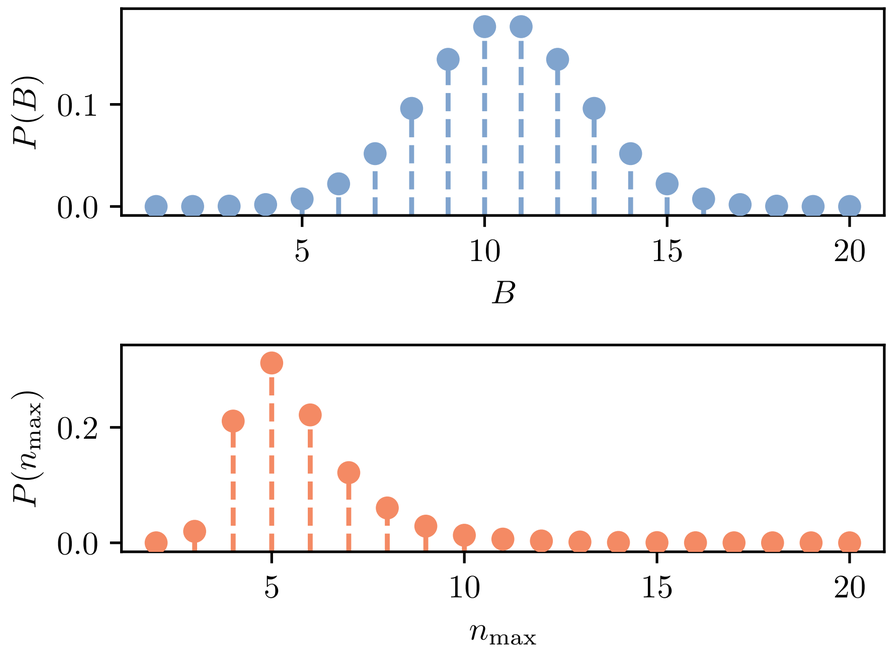
$\Omega(\mathbf{e}\,\vert\,\mathbf{b})$: the number of block count matrices compatible with the given partition $\mathbf{b}$.

$\Omega(\mathbf{A}\,\vert\,\mathbf{e},\mathbf{b})$: the number of graphs that satisfy the constraints of $\mathbf{e}$ and $\mathbf{b}$.

The microcanonical stochastic block model

Updating the partition prior

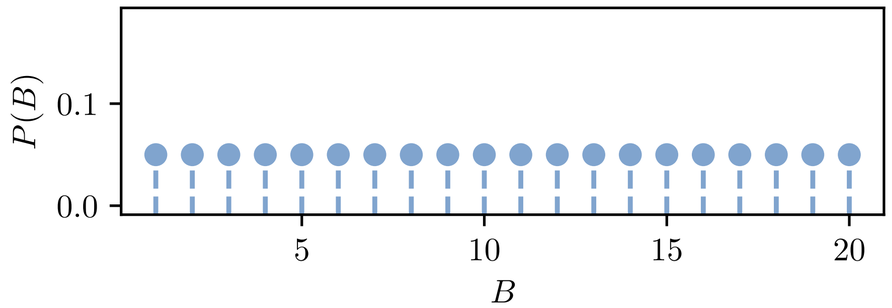
The original prior $P(\mathbf{b}) = \Omega(\mathbf{b})^{-1}$ implies a particular characteristic group size.



Introduce a new prior fixes these biases:

$$P(\mathbf{b}) = P(\mathbf{b}\,|\,\mathbf{n})P(\mathbf{n}\,|\,B)P(B),$$

where B is the number of groups and $\mathbf{n}$ is a vector of group sizes.

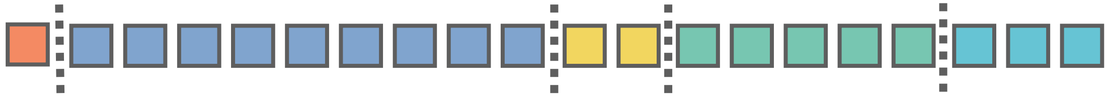


The microcanonical stochastic block model

Updating the partition prior

$$B = 5$$

$$\mathbf{n} = (1,9,2,5,3)$$

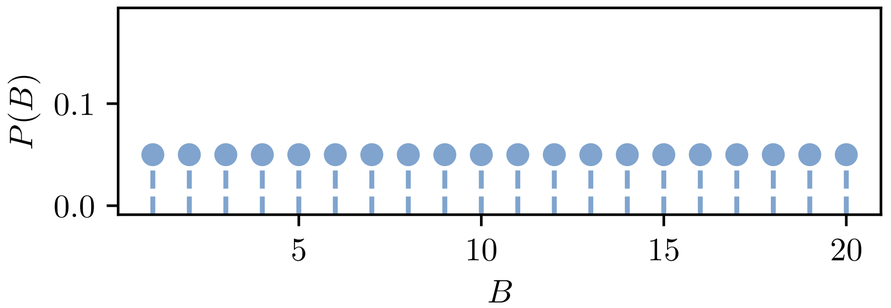


$$\mathbf{b} = (0,1,1,1,2,1,1,2,3,3,1,1,4,3,4,4,3,1,3,1)$$

New prior fixes these biases:

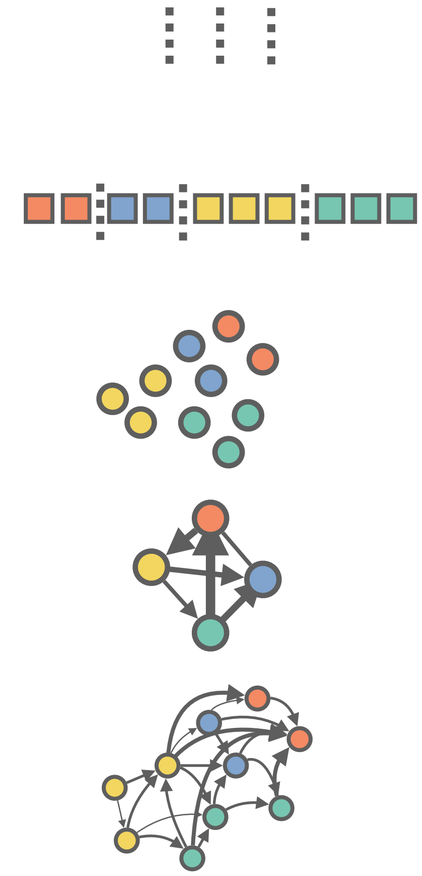
$$P(\mathbf{b}) = P(\mathbf{b}\,|\,\mathbf{n})P(\mathbf{n}\,|\,B)P(B),$$

where B is the number of groups and $\mathbf{n}$ is a vector of group sizes.



The microcanonical stochastic block model

	MDL encoding	Generative model
Num. groups	Transmit an integer between 0 and N	$P(B) = \frac{1}{N}$
Group sizes	Transmit an integer between 0 and $\Omega(\mathbf{n}\,\vert\,B)$	$P(\mathbf{n}) = \frac{1}{\Omega(\mathbf{n}\,\vert\,B)}$
Partition	Transmit an integer between 0 and $\Omega(\mathbf{b})$	$P(\mathbf{b}\,\vert\,\mathbf{n}) = \frac{1}{\Omega(\mathbf{b}\,\vert\,\mathbf{n})}$
Block count matrix	Transmit an integer between 0 and $\Omega(\mathbf{e}\,\vert\,\mathbf{b})$	$P(\mathbf{e}\,\vert\,\mathbf{b}) = \frac{1}{\Omega(\mathbf{e}\,\vert\,\mathbf{b})}$
Graph	Transmit an integer between 0 and $\Omega(\mathbf{A}\,\vert\,\mathbf{e},\mathbf{b})$	$P(\mathbf{A}\,\vert\,\mathbf{e},\mathbf{b}) = \frac{1}{\Omega(\mathbf{A}\,\vert\,\mathbf{e},\mathbf{b})}$

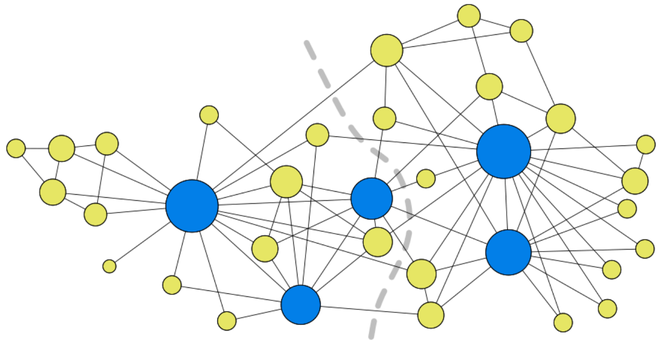


Model variants

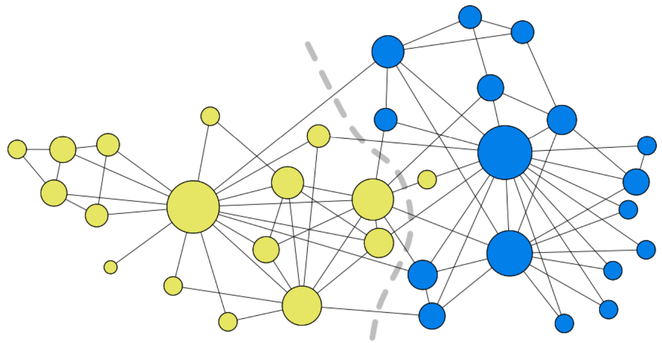
Degree correction

Degree correction considers deviations from randomness after taking into account the degrees of each node.

Karate club

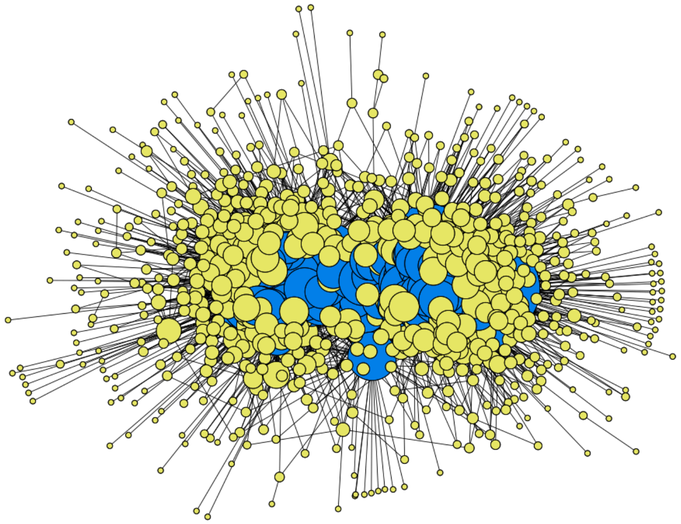


(a) Without degree correction

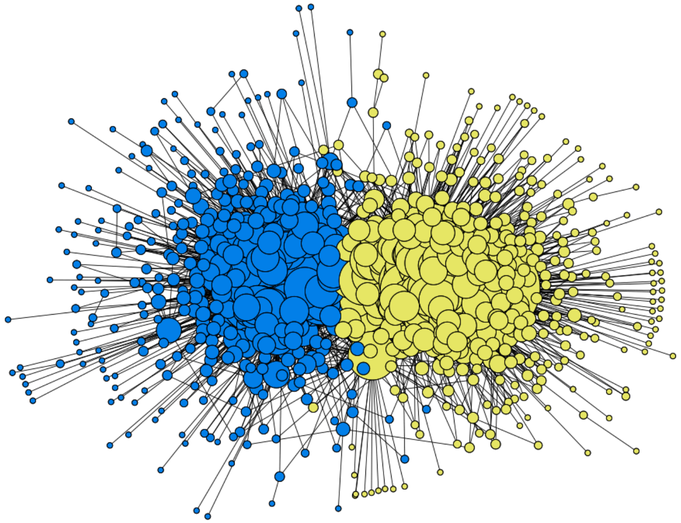


(b) With degree-correction

Political books



(a) Without degree-correction



(b) With degree-correction

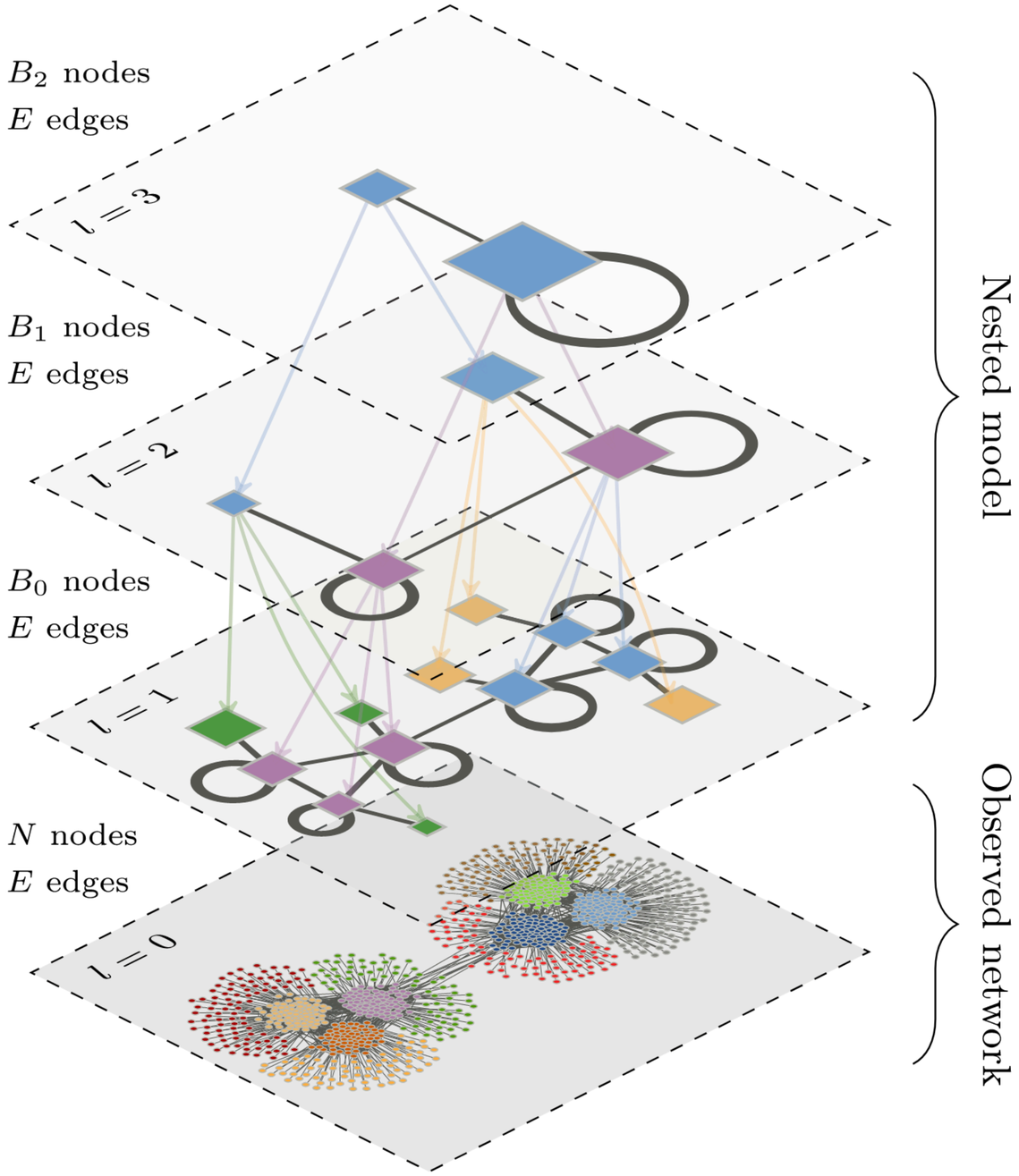
Karrer and Newman. "Stochastic blockmodels and community structure in networks," *PRE (2011)*

Model variants

Nested models

Nested models allow for an improved resolution limit, compared to modularity maximization.

Also offer a richer characterization of the system.



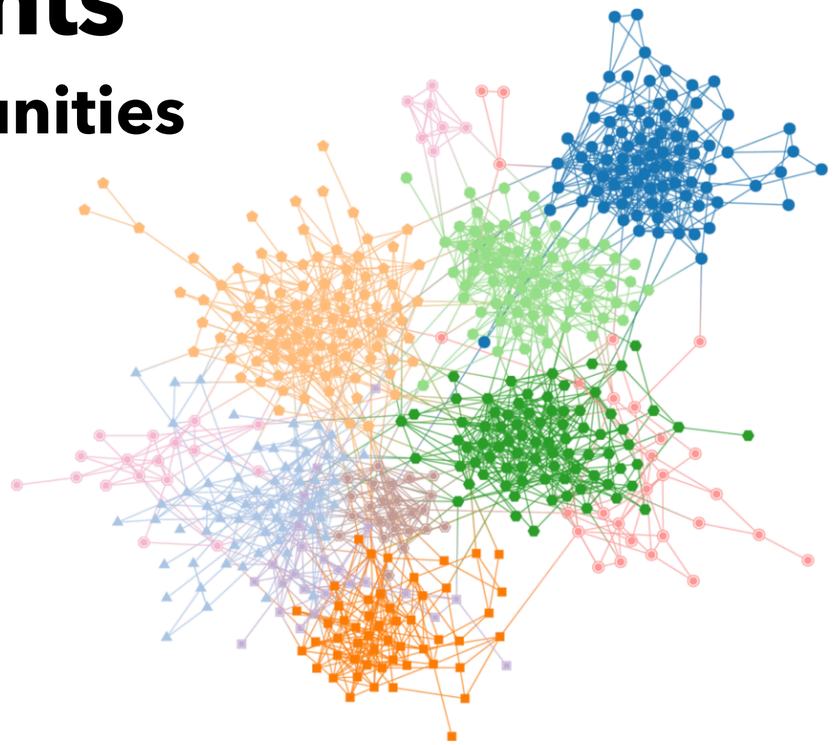
Peixoto. "Hierarchical block structures and high-resolution model selection in large networks," *PRX (2014)*

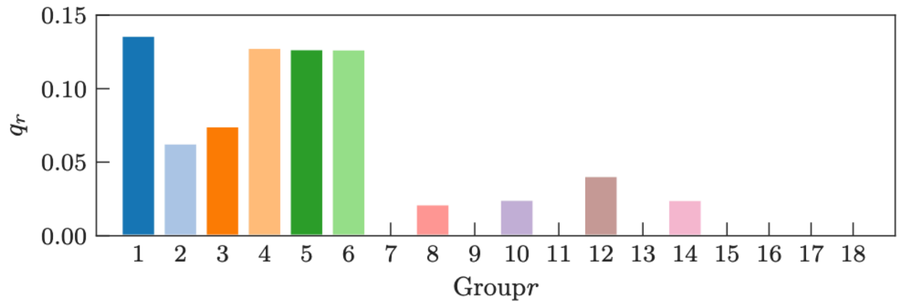
Model variants

Assortative communities

The non-uniform planted partition model assumes a difference between *internal* and *external* edges, not a general mixing pattern.

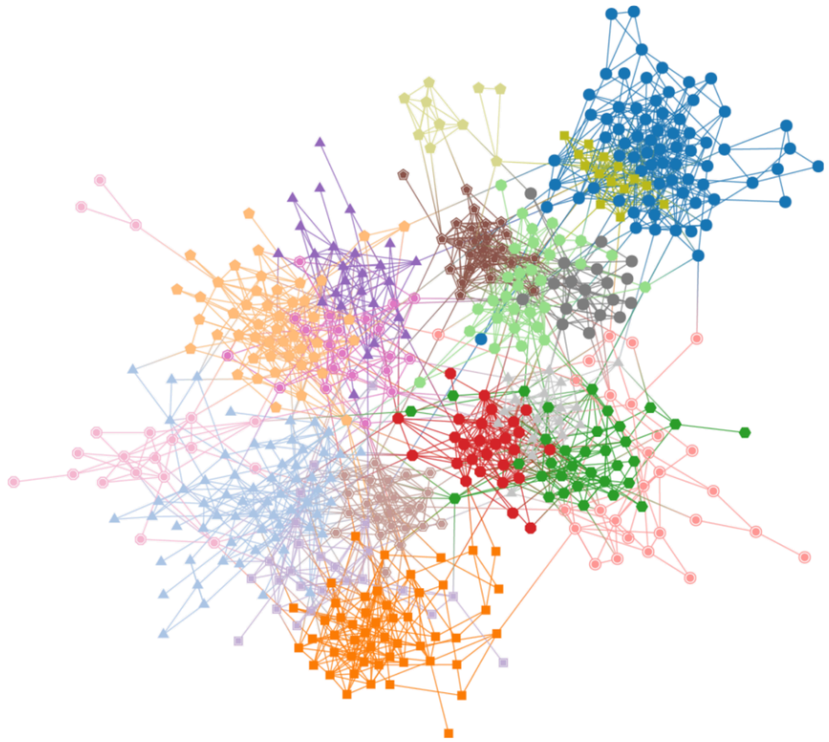
(a) PP (non-uniform)

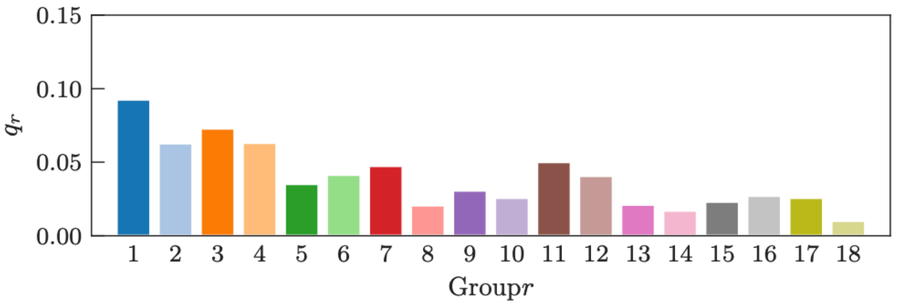




$\Sigma = 8944.09$ (nats), $Q = 0.765$

(b) Nested DC-SBM





$\Sigma = 8775.82$ (nats), $Q = 0.706$

Peixoto. "Statistical inference of assortative community structures," *PRR (2020)*

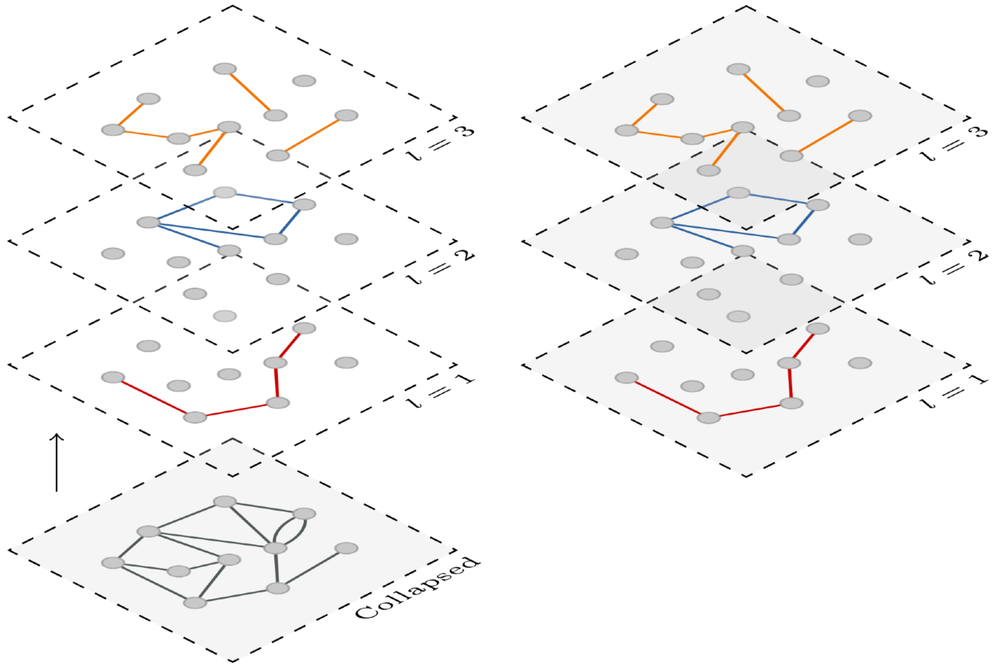
Model variants

Multilayer networks

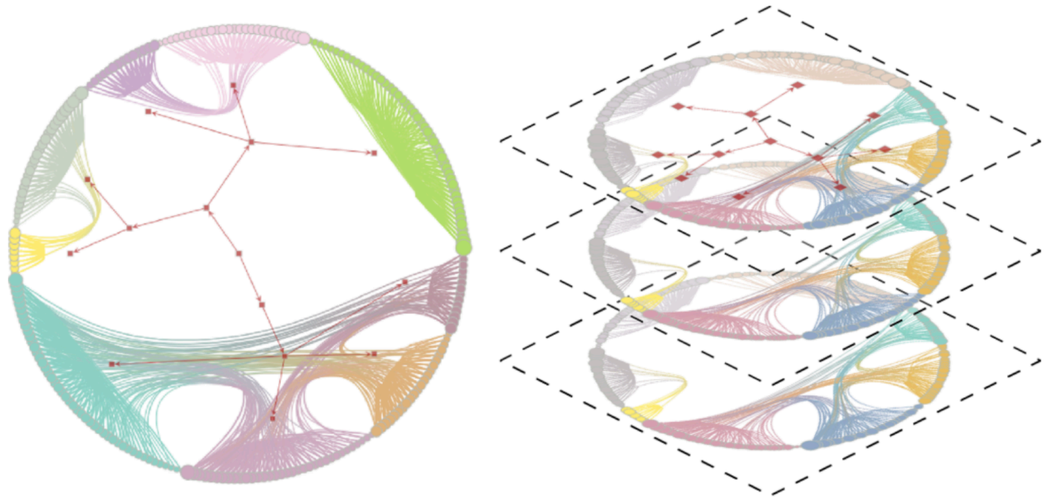
Model networks with edge layers $\{A_{ij}^{l}\}$ in two ways:

1. Model the collapsed network first, then randomly distribute edges
2. Model each layer independently.

MDL allows model selection between these options.



Edge covariates | Independent layers



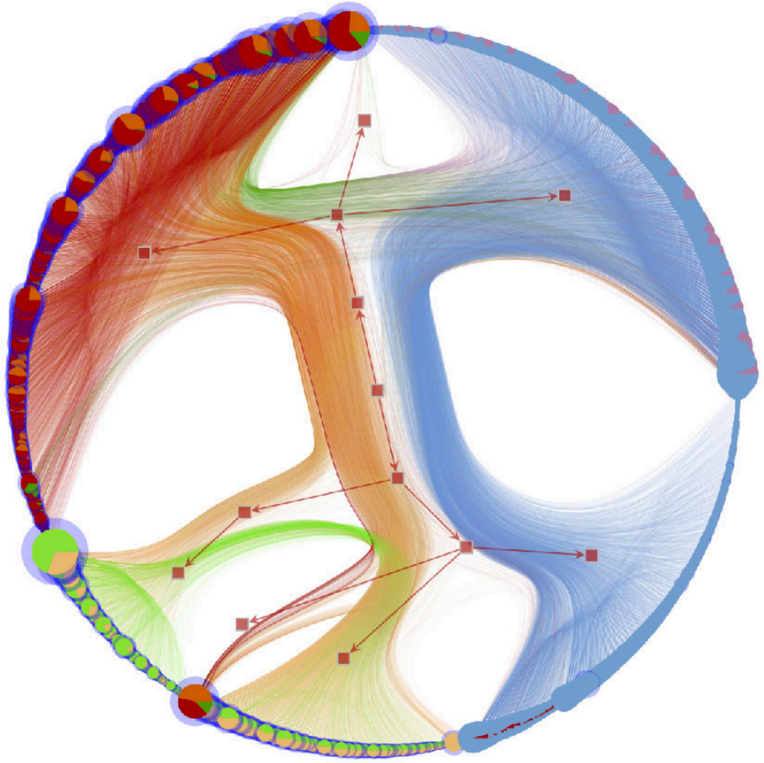
(a) $\Lambda = 1$ (b) $\log_{10}\Lambda \approx -51$

Peixoto. "Inferring the mesoscale structure of layered, edge-valued, and time-varying networks," *PRE (2015)*

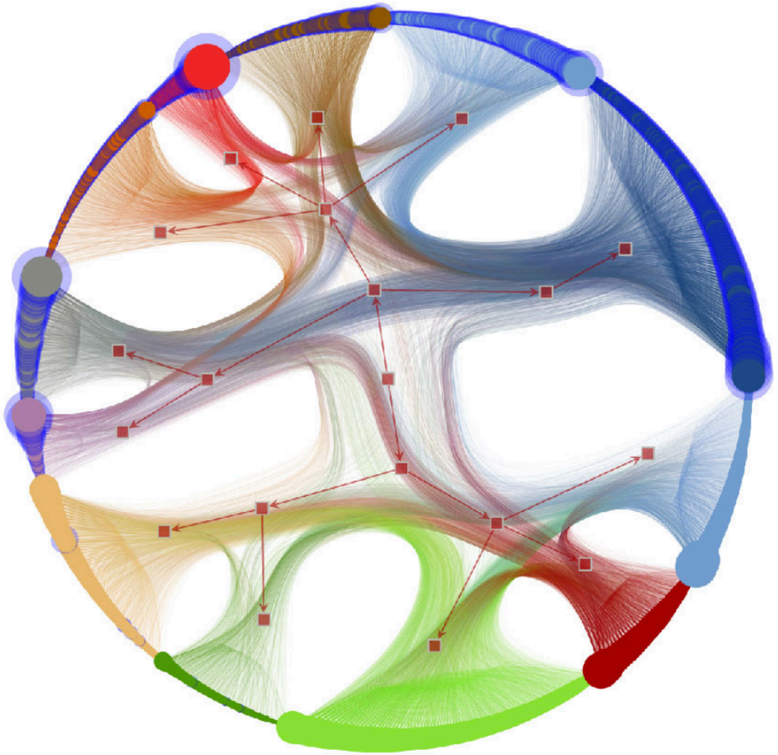
Model variants

Overlapping groups

Allow nodes to participate in multiple groups, and hence their neighbors may come from these multiple affiliations.



$B = 7$, overlapping, degree-corrected, $\Lambda = 1$



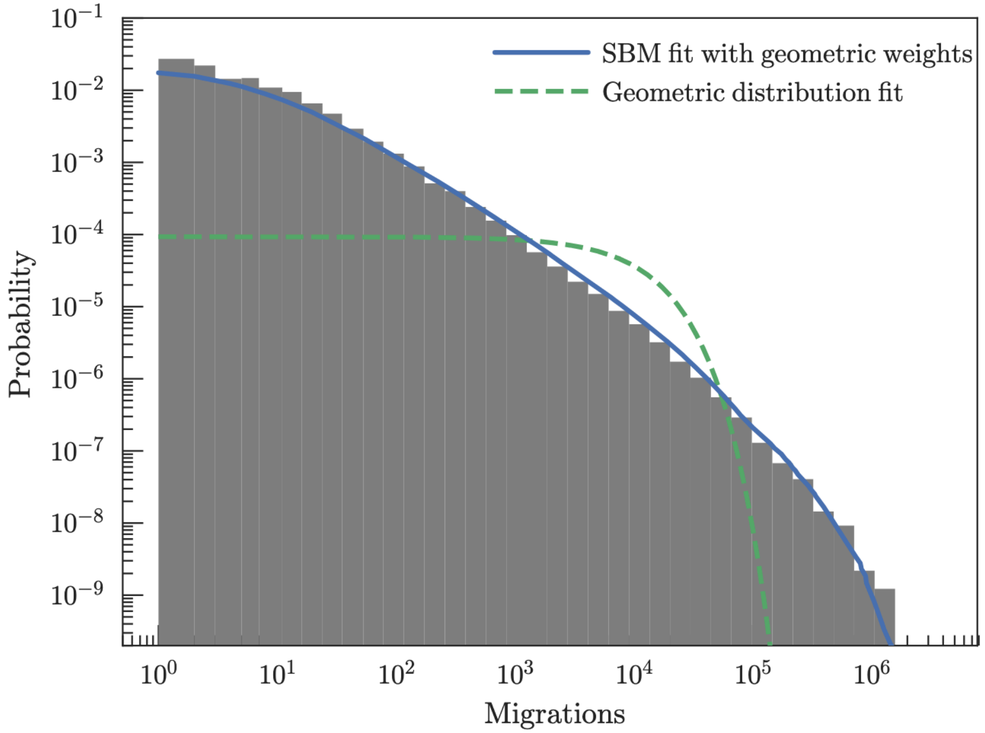
$B = 12$, nonoverlapping, degree-corrected, $\log_{10} \Lambda \simeq -747$

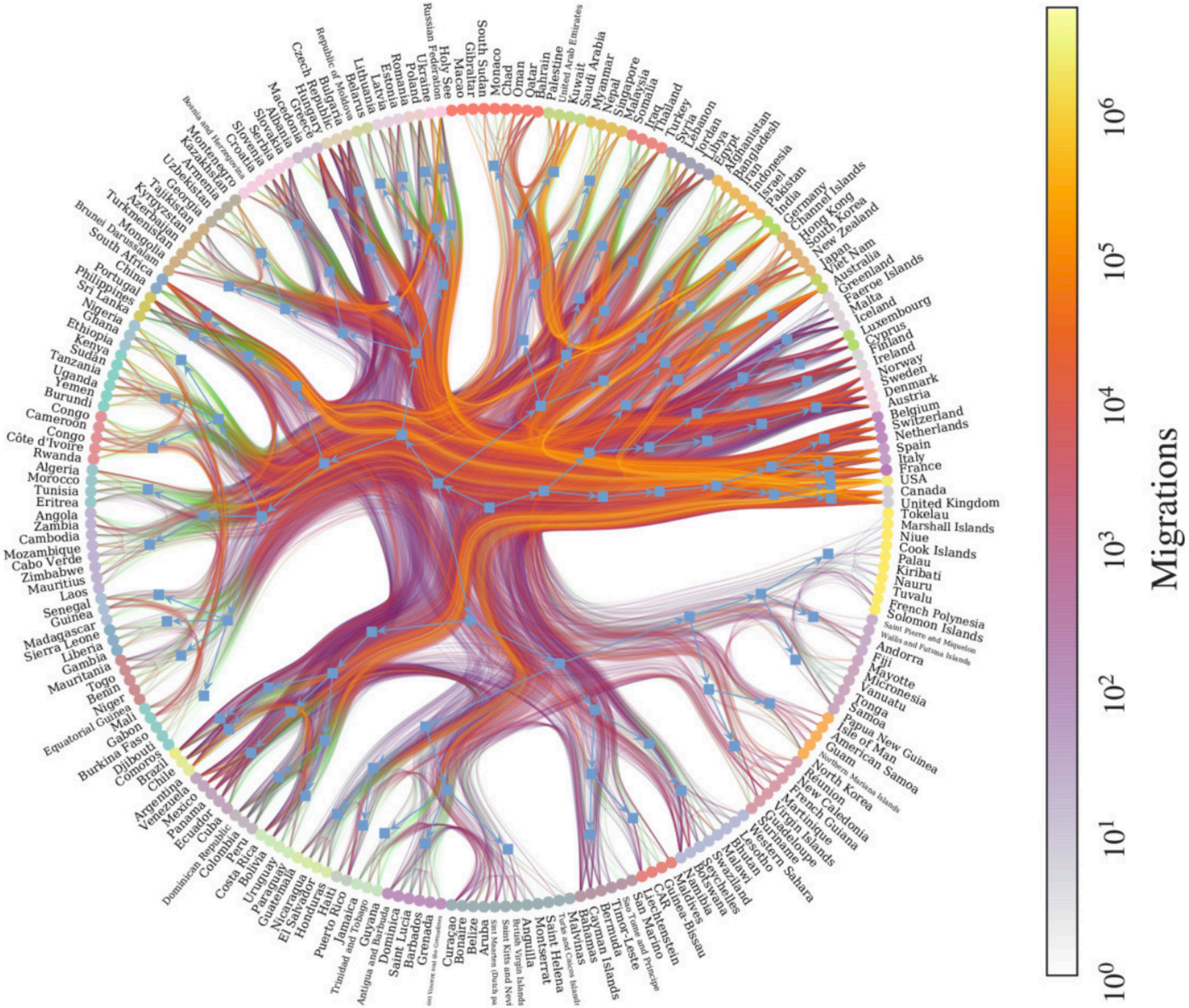
Peixoto. "Model selection and hypothesis testing for large-scale network models with overlapping groups," *PRX (2015)*

Model variants

Edge weights

Edge weights represent edge strength.
Layers represent edge classification.



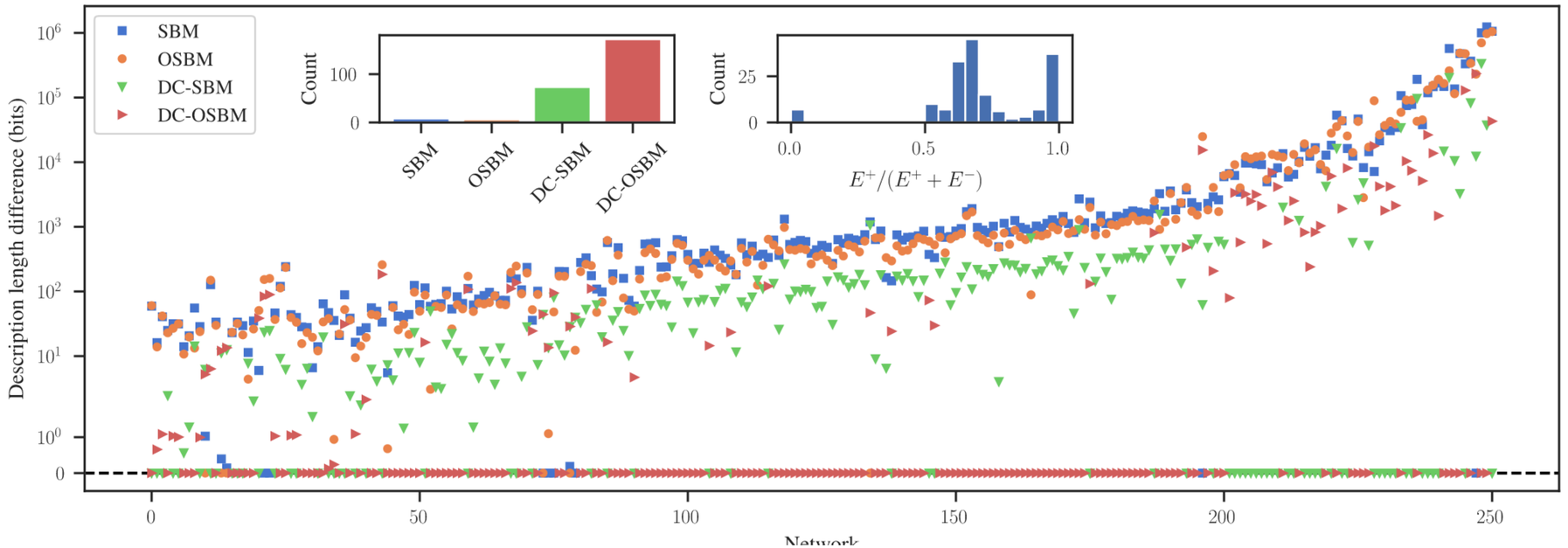


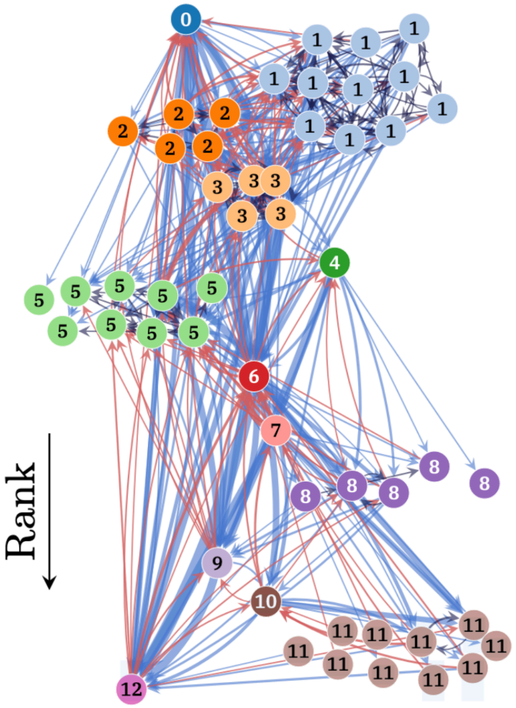
Peixoto. "Nonparametric weighted stochastic blockmodels," *PRE (2018)*

Model variants

Hierarchies and ranking

If blocks can be ordered such that edges flow in one direction, this is evidence of hierarchies in data.





$\Sigma = 2912.1$ bits

Peixoto. "Ordered community detection in directed networks," *PRE (2022)*

See notebook: `gt-statistical-inference.ipynb`

Clustering bipartite graphs

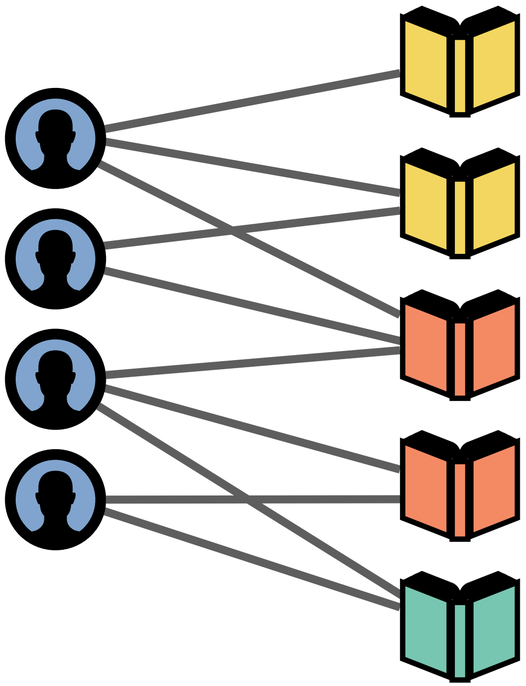
Bipartite graphs are everywhere!

People and _____

People vs. websites they visit

People vs. books they read

People vs. places they visit



Higher order systems

Scientific collaboration:

- Authors vs. papers / grants

Topic modeling:

- Words vs. documents
- Songs vs. playlists

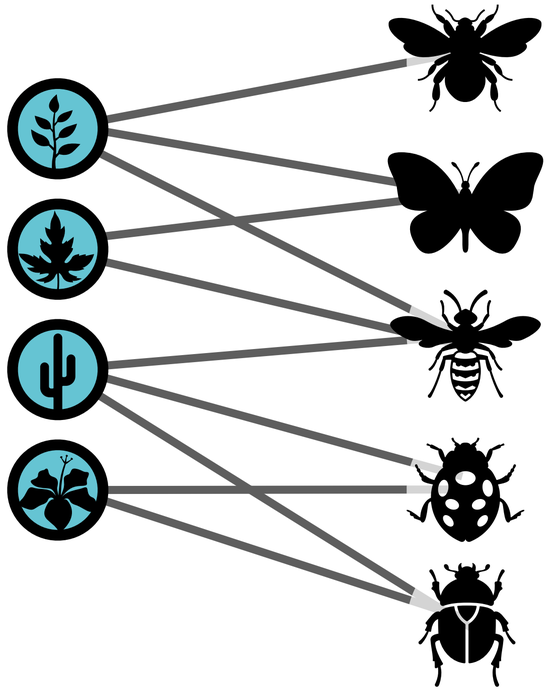
Social groups:

- Companies vs. board members
- Friends vs. friend groups
- Characters vs. scenes

Mutualistic interactions

Plants vs. pollinators

Microbes vs. hosts



Example bipartite analysis

Finding biomarkers of sea star wasting disease

Microbial dysbiosis precedes signs of sea star wasting disease in wild populations of *Pycnopodia helianthoides*

Andrew R. McCracken[1,2*], Blair M. Christensen[1,3], Daniel Munteanu[1,2], B. K. M. Case[1,4], Melanie Lloyd[2], Kyle P. Herbert[5] and Melissa H. Pespeni[1,2*]

[1]Quantitative and Evolutionary STEM Training (QuEST) Program, University of Vermont, Burlington, VT, United States, [2]Department of Biology, University of Vermont, Burlington, VT, United States, [3]Department of Plant and Soil Science, University of Vermont, Burlington, VT, United States, [4]Department of Computer Science, University of Vermont, Burlington, VT, United States, [5]Alaska Department of Fish and Game, Douglas, AK, United States

Example bipartite analysis

Finding biomarkers of sea star wasting disease

- Disease affecting sea stars amidst rise global sea temperatures
- No specific cause of the disease, meaning no obvious cure
- Approach: examine the microbiomes of healthy and sick sea stars

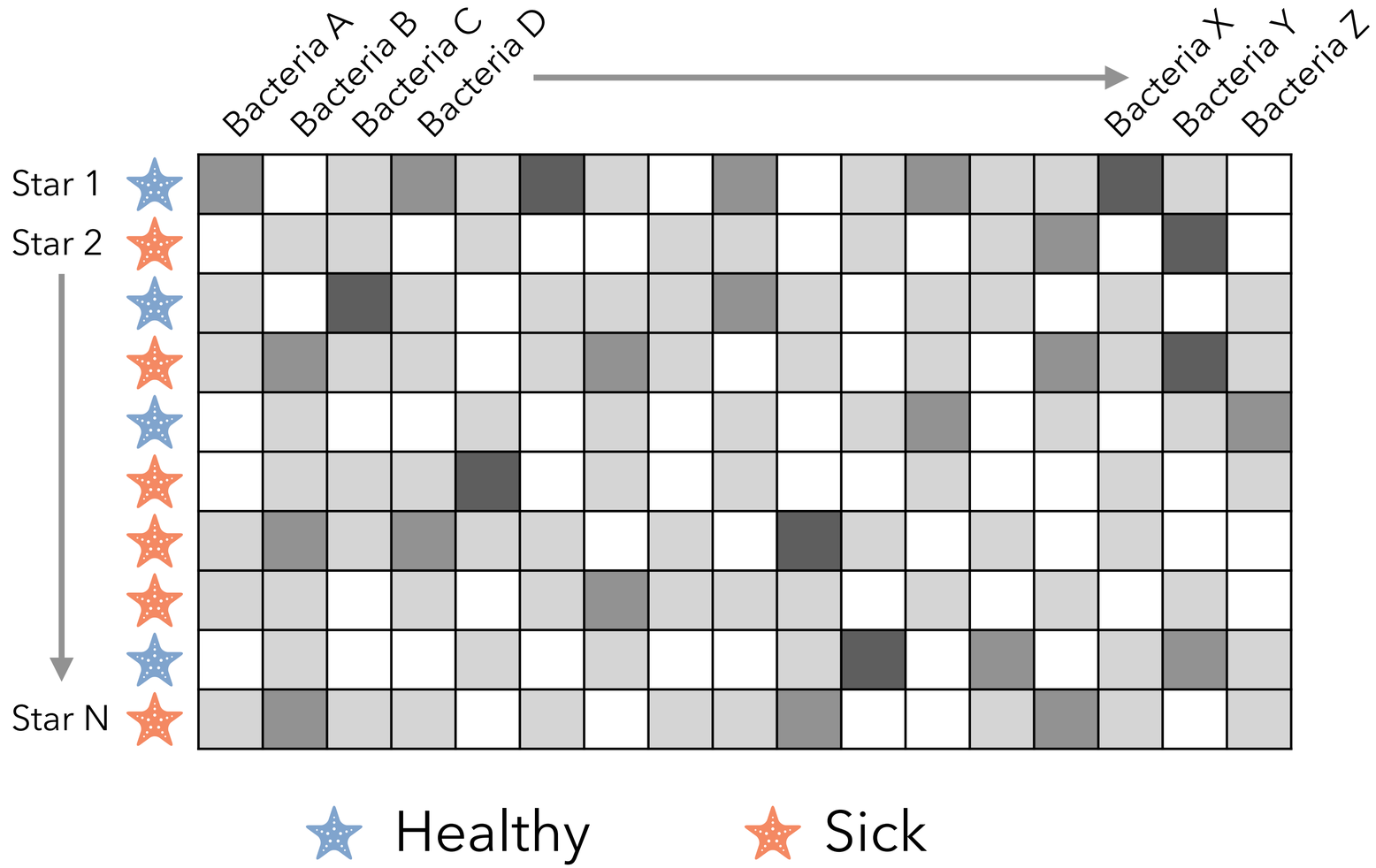
McCracken et al. "Microbial dysbiosis precedes signs of sea star wasting disease in wild populations of *Pycnopodia helianthoides*," (2023)

Example bipartite analysis

Finding biomarkers of sea star wasting disease

Data: Measured the concentrations of different bacterial strains on each sea stars using biology witchcraft

Idea: perform clustering analysis on the sea stars to see if there are similarities between the communities and the sick/healthy labels.



Example bipartite analysis

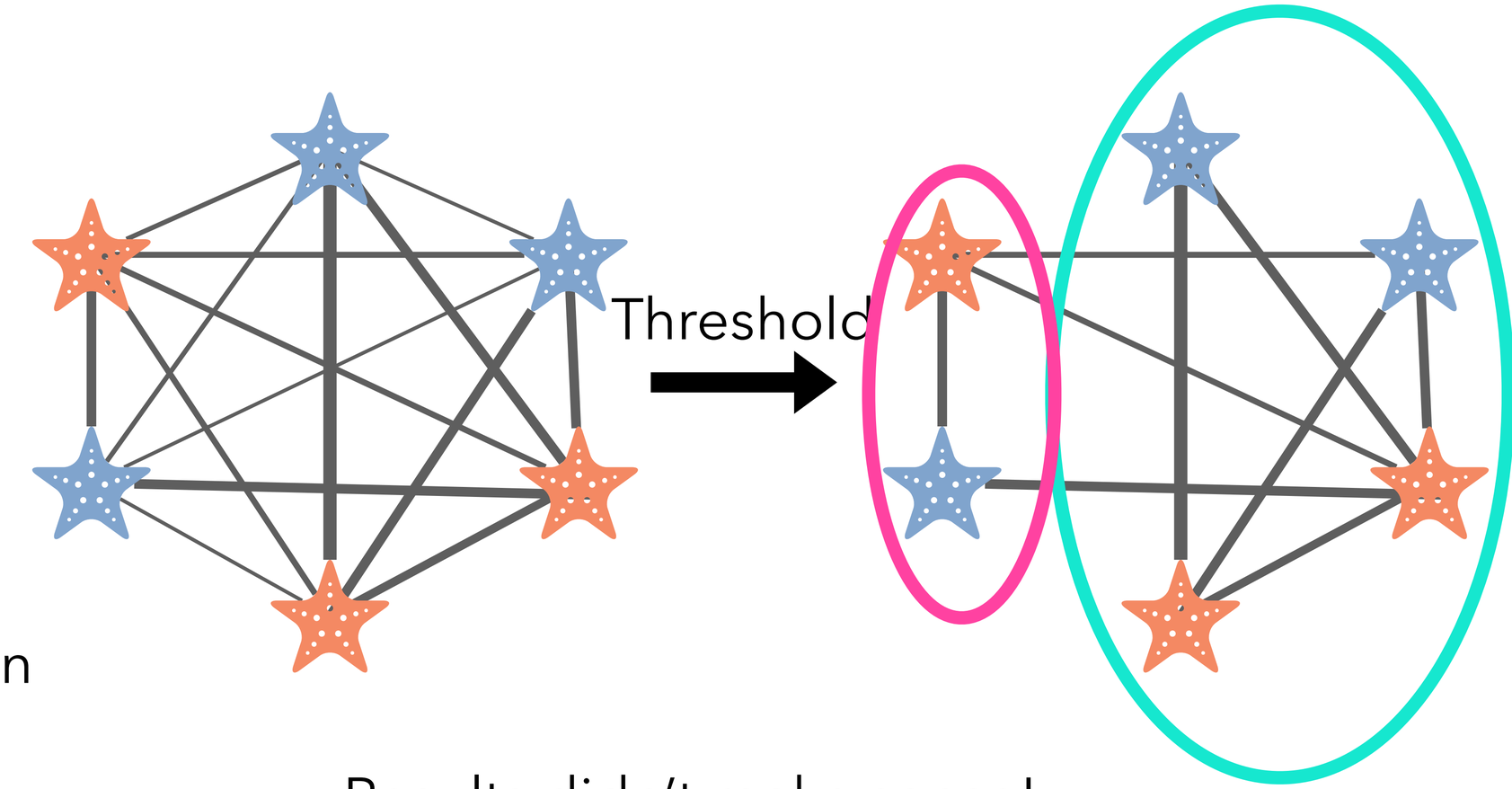
Finding biomarkers of sea star wasting disease

First attempt:

1. Compute a similarity score between all pairs of sea stars and from a network.

2. Threshold the network because it is dense.

3. Do community detection using modularity maximization.



Results didn’t make sense!

Example bipartite analysis

Finding biomarkers of sea star wasting disease

First attempt:

1. Compute
score betwe
sea stars an
network.

2. Threshol
because it i

3. Do comm
using modu
maximization.

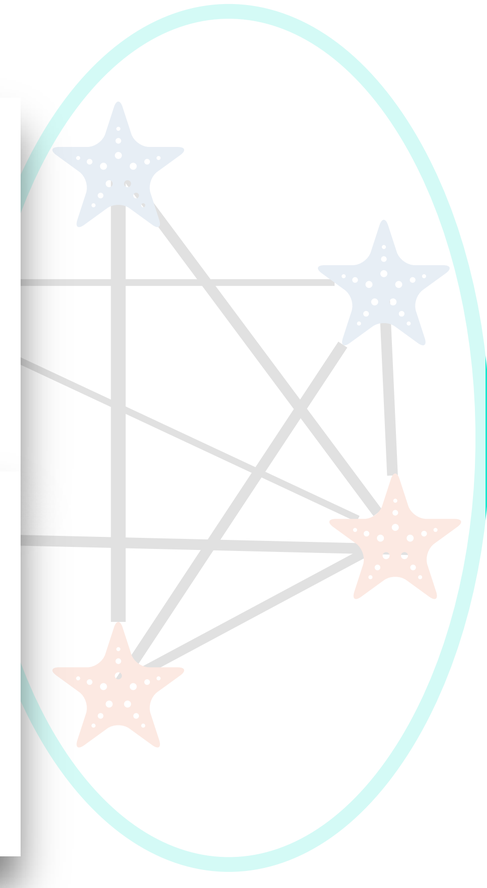
PHYSICAL REVIEW E **101**, 062302 (2020)

Thresholding normally distributed data creates complex networks

George T. Cantwell,[1,*] Yanchen Liu,[2] Benjamin F. Maier,[3,4] Alice C. Schwarze,[5] Carlos A. Serván,[6] Jordan Snyder,[7,8] and Guillaume St-Onge[9,10]

Network data sets are often constructed by some kind of thresholding procedure. The resulting networks frequently possess properties such as heavy-tailed degree distributions, clustering, large connected components, and short average shortest path lengths. These properties are considered typical of complex networks and appear in many contexts, prompting consideration of their universality. Here we introduce a simple model for correlated relational data and study the network ensemble obtained by thresholding it. We find that some, but not all, of the properties associated with complex networks can be seen after thresholding the correlated data, even though the underlying data are not "complex." In particular, we observe heavy-tailed degree distributions, a large numbers of triangles, and short path lengths, while we do not observe nonvanishing clustering or community structure.

DOI: 10.1103/PhysRevE.101.062302



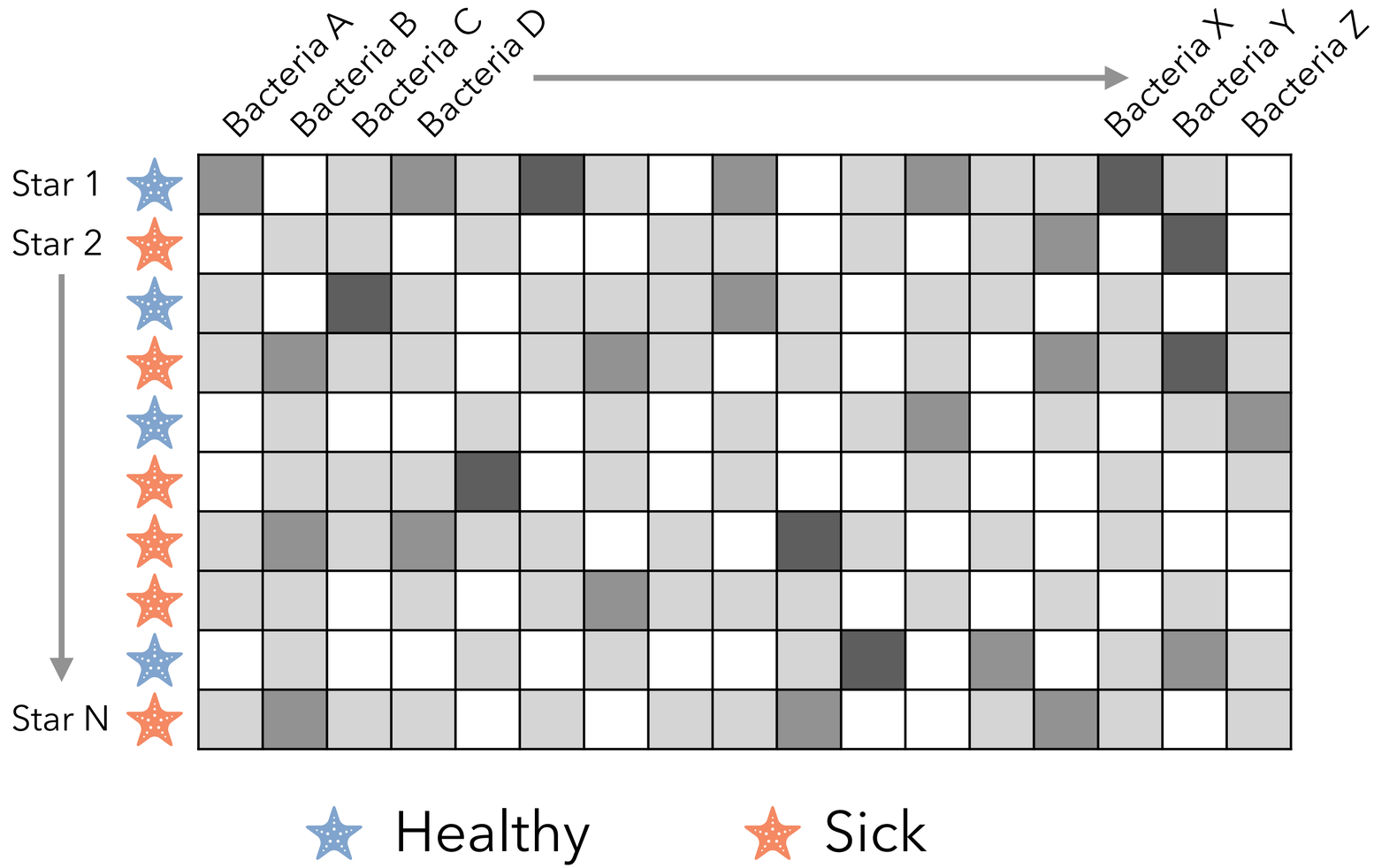
Results didn't make sense!

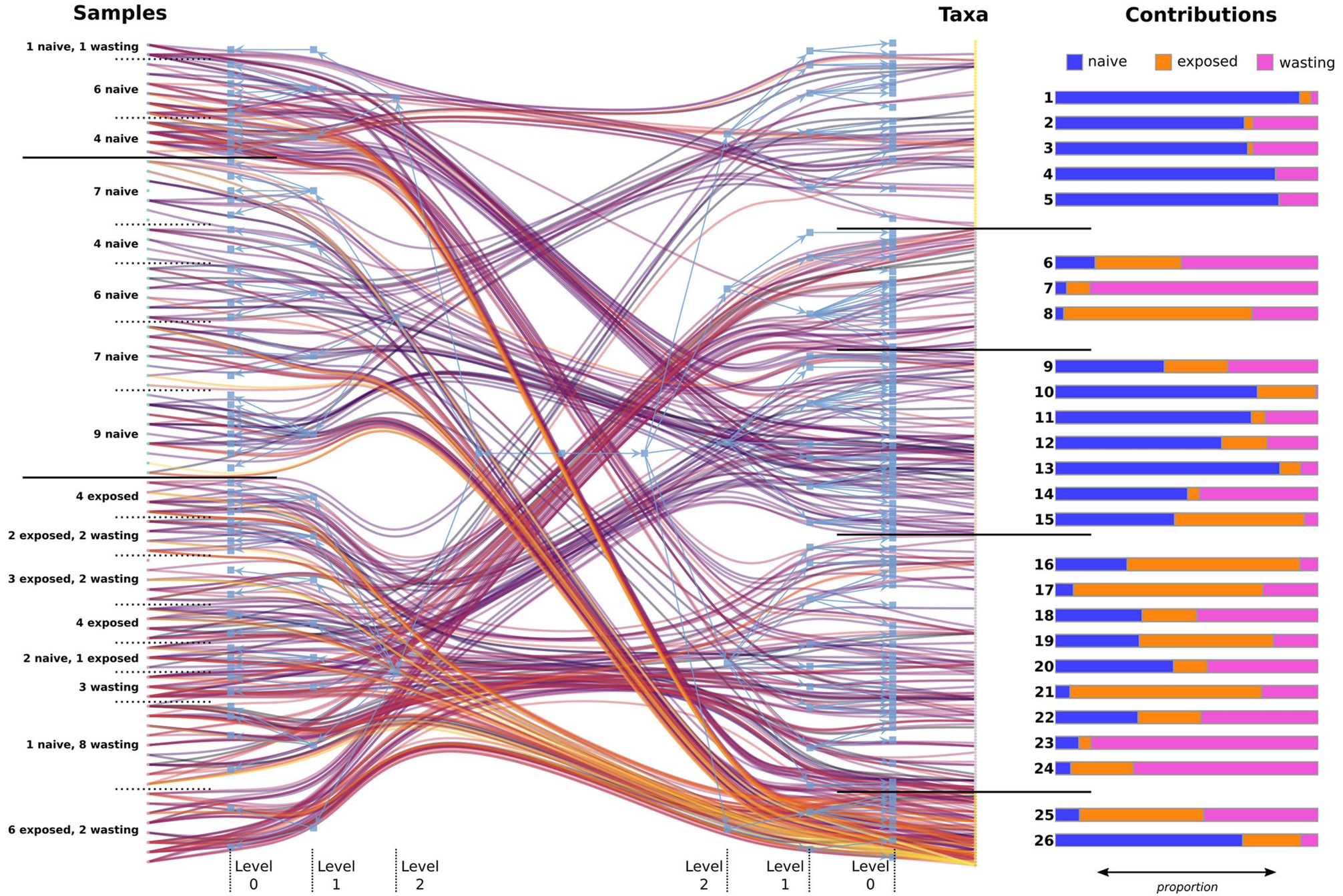
Example bipartite analysis

Finding biomarkers of sea star wasting disease

New idea: perform clustering analysis directly on the bipartite graph of sea stars and microbes.

This matrix is just the adjacency matrix of a *weighted* bipartite graph.





McCracken et al. "Microbial dysbiosis precedes signs of sea star wasting disease in wild populations of *Pycnopodia helianthoides*," (2023)

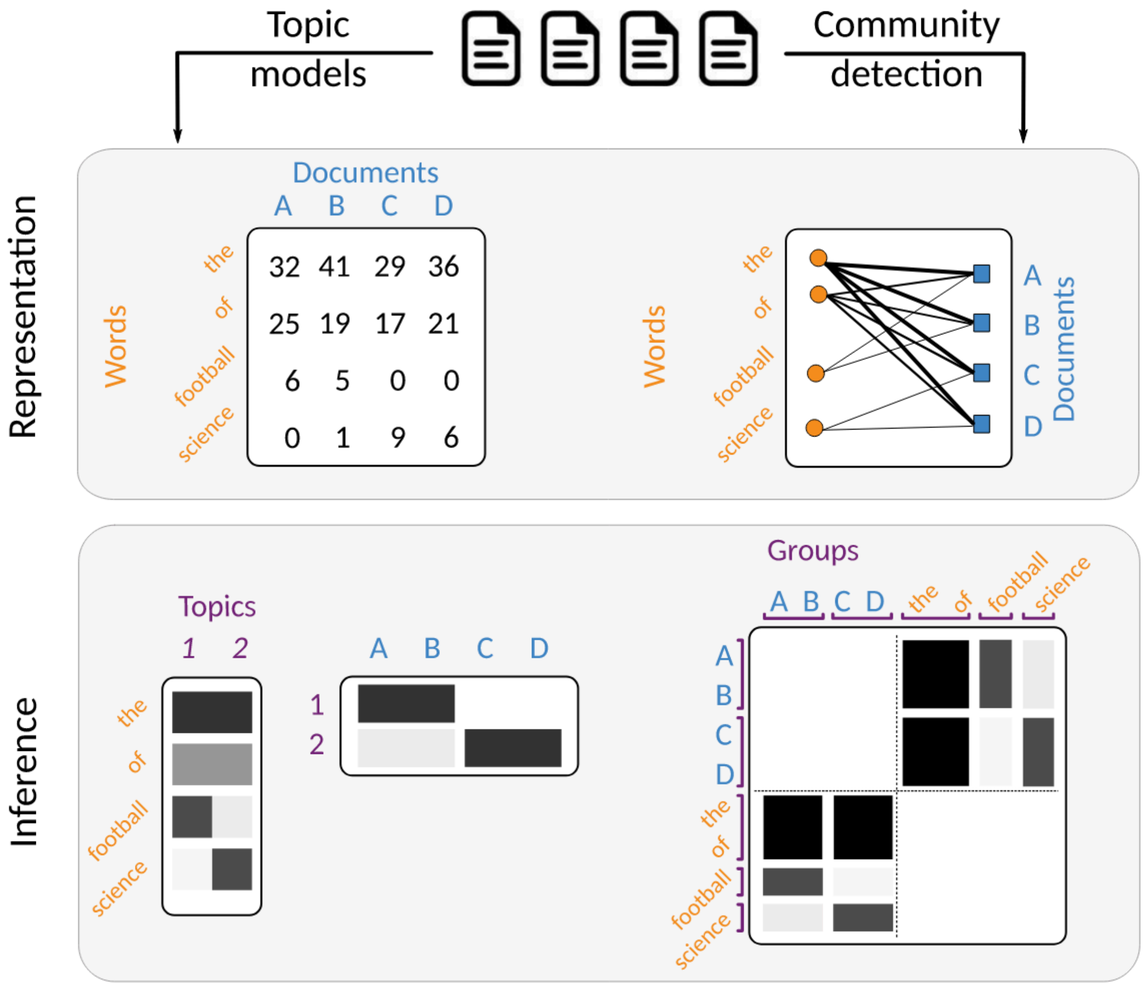
Topic modeling

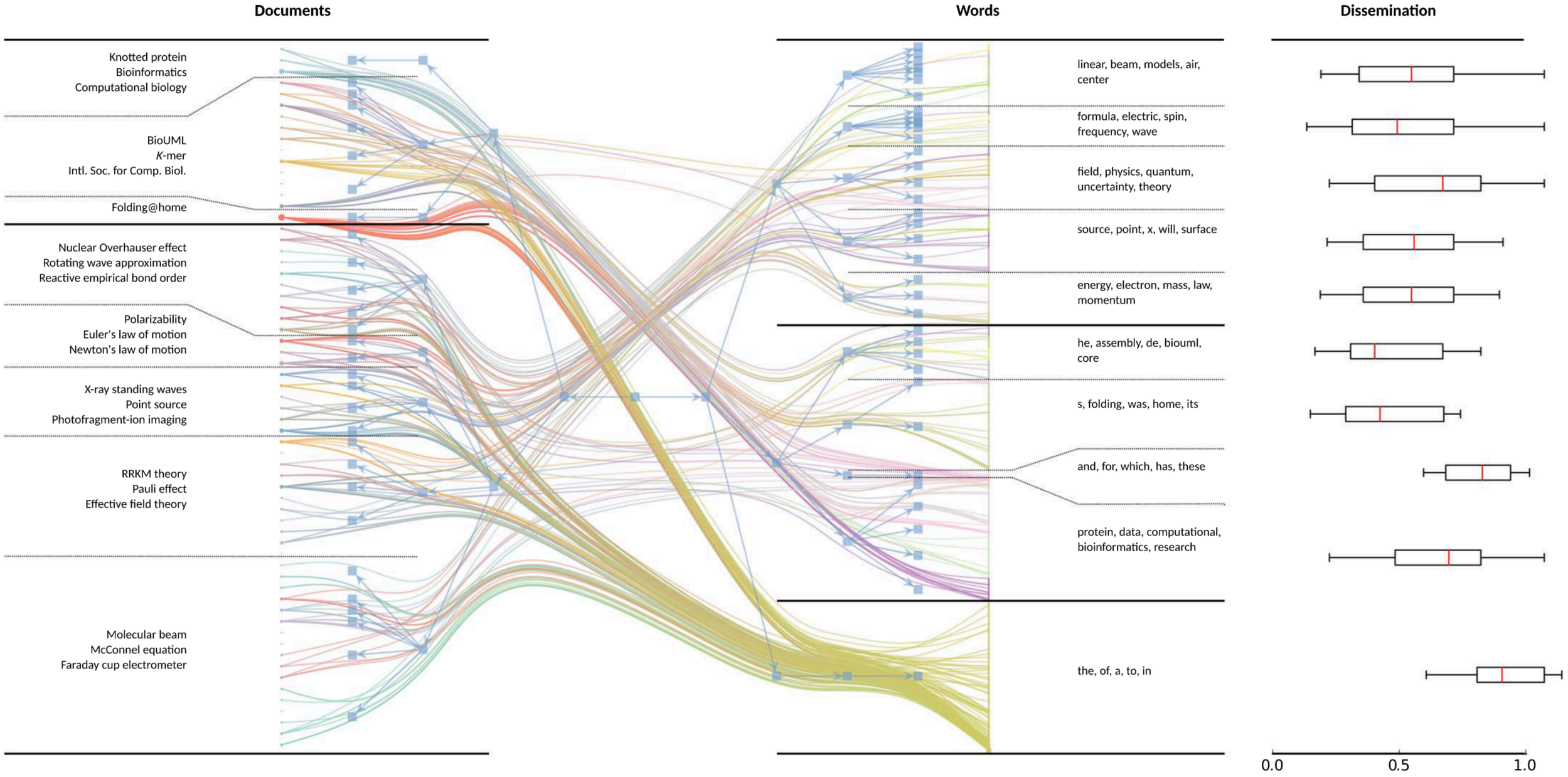
As SBM inference

Bipartite network of words and documents.

All that's needed is to modify the prior to enforce the constraint that certain nodes can *never* be placed in the same group:

$$P(\mathbf{b}) = P(\mathbf{b}^{\text{word}})P(\mathbf{b}^{\text{doc}})$$





Gerlach et al.. "A network approach to topic models" *Science Advances (2018)*

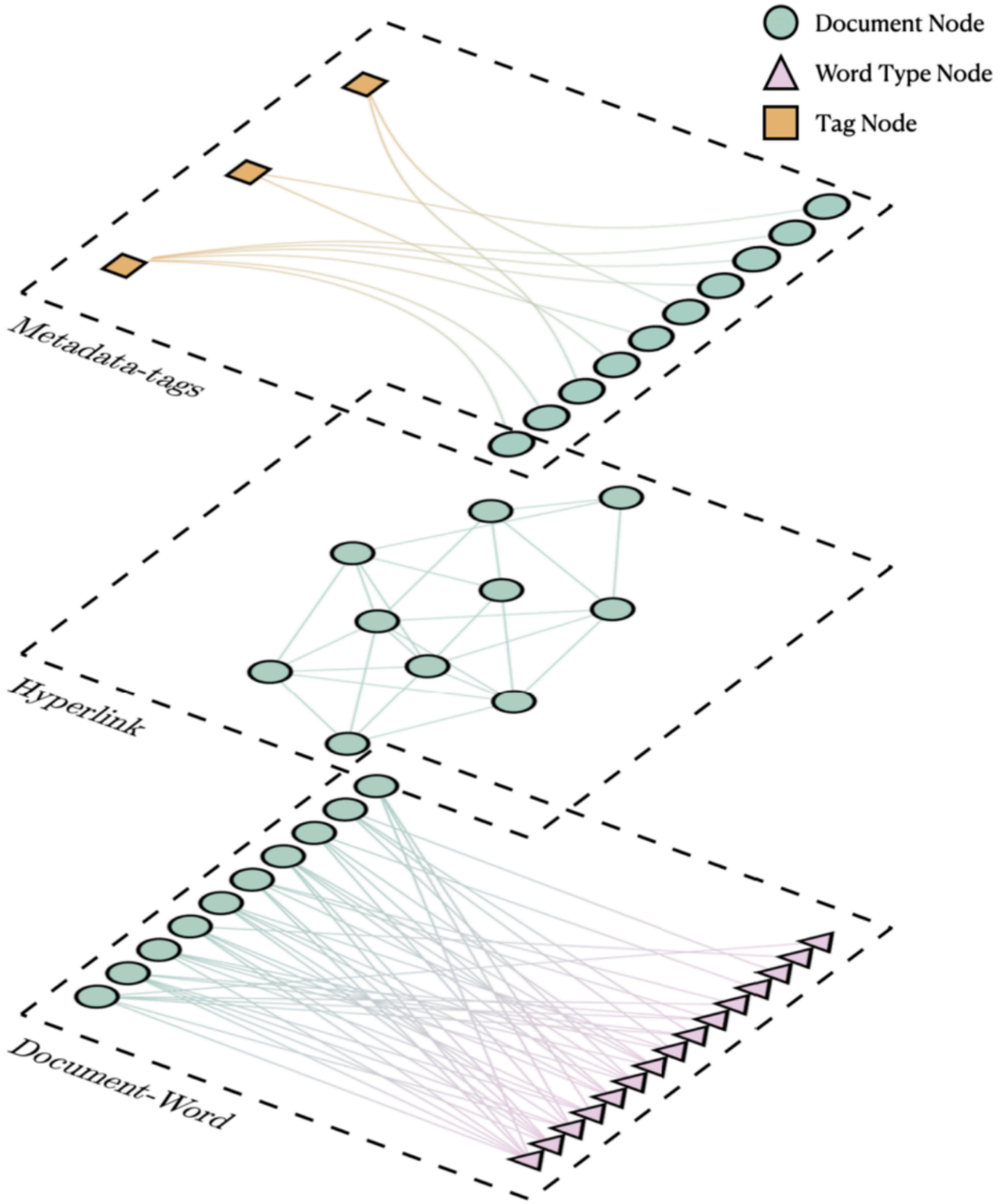
Topic modeling

As SBM inference

Include metadata as additional layers of the network.

Metadata tags for another bipartite network.

Relational data forms networks among node subsets, e.g., hyperlinks



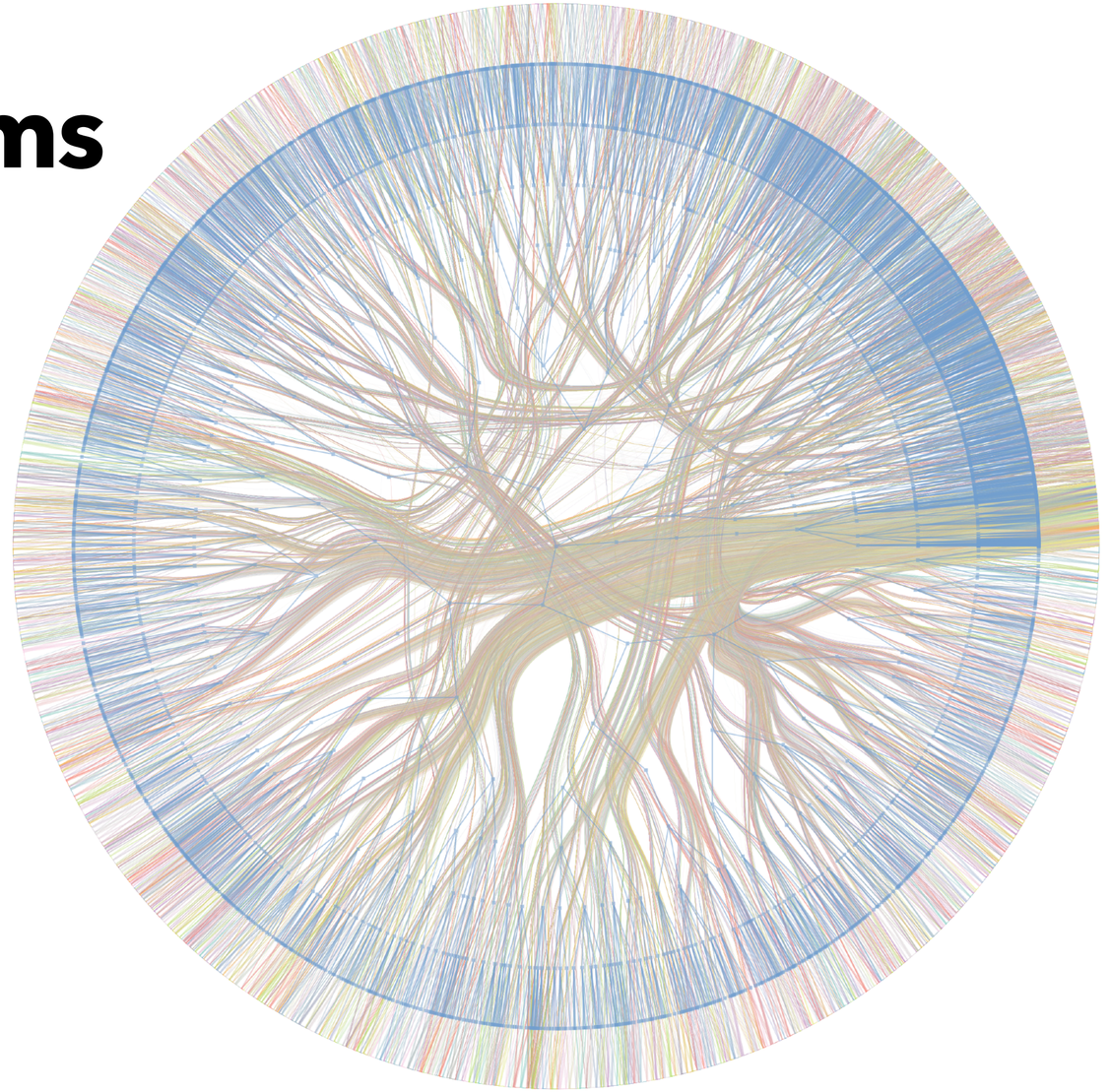
Hyland et al. “Multilayer networks for text analysis with multiple data types” *EPJ Data Science (2021)*

Recommendation algorithms

As SBM inference

Also useful for other things, like recommendation systems, e.g. Spotify

- Bipartite network of songs and playlists
- Bipartite network of albums and songs
- Bipartite network of albums and artists
- Bipartite network of songs and artists



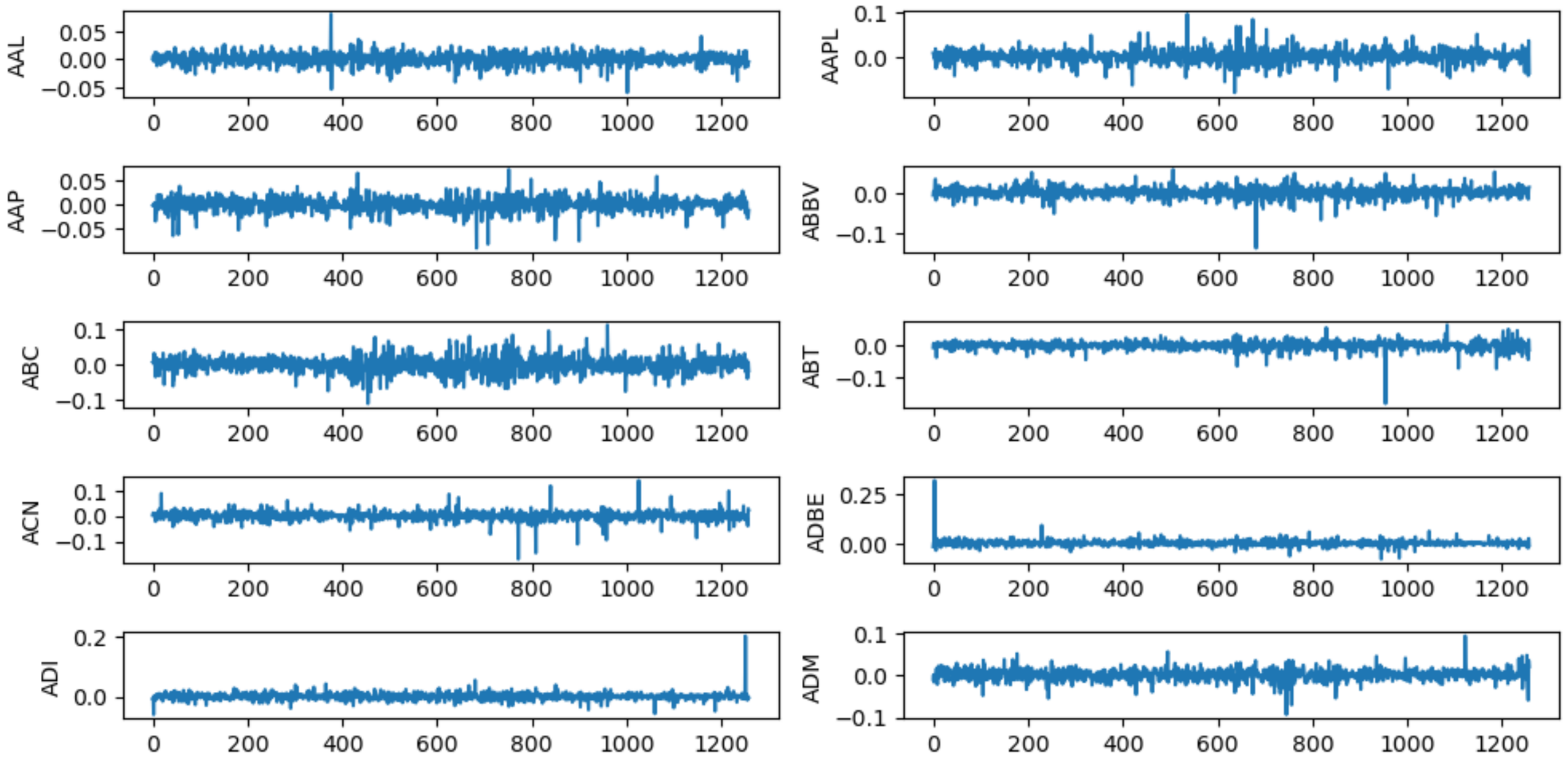
Spotify Million Playlist Dataset

Spotify Million Playlist Dataset (https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge)

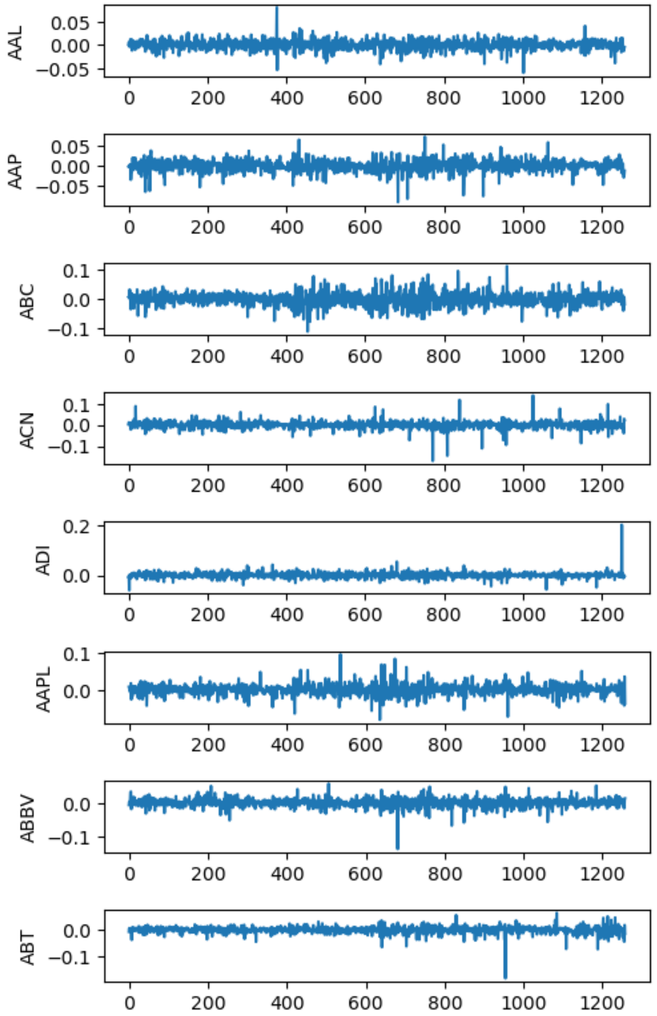
See notebook: `gt-statistical-inference.ipynb`

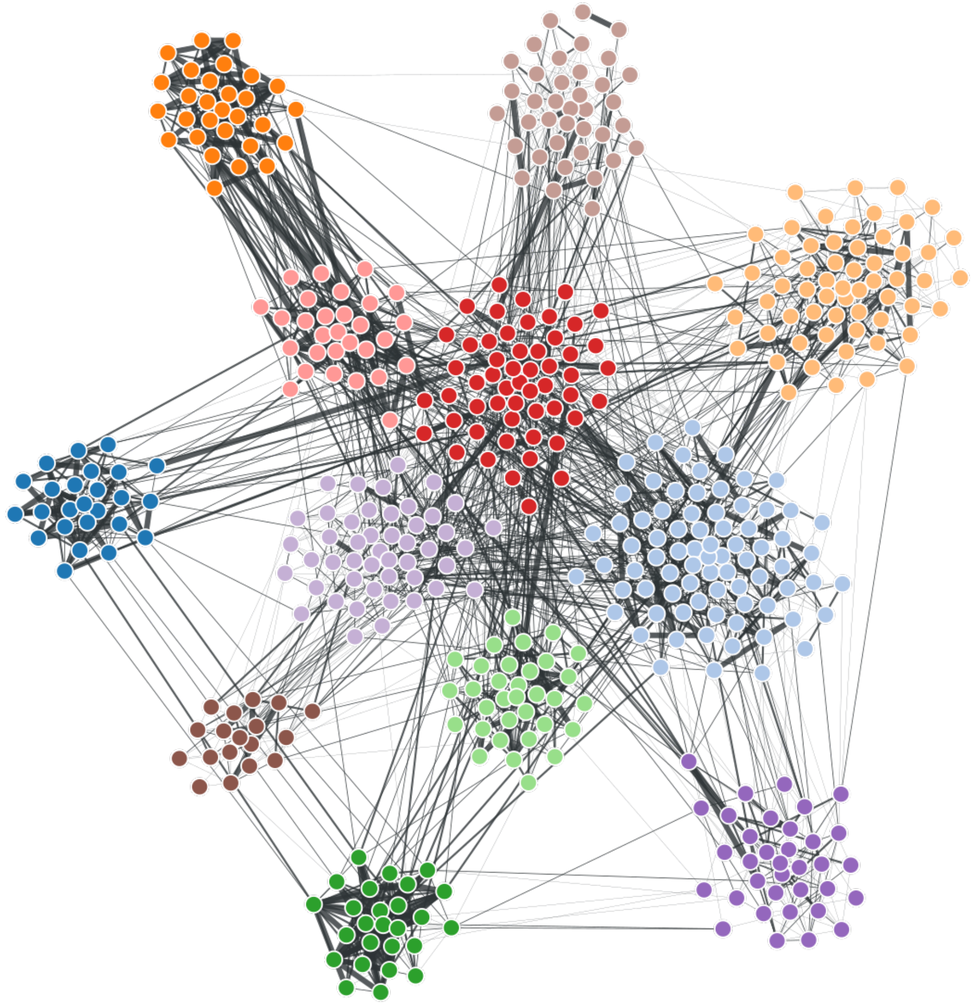
Learning networks and dynamics from multiple measurements and time series

Network reconstruction from time series



Network reconstruction from time series





Network reconstruction from time series

Define a generative model that samples dynamics data from a *known* dynamical process with a closed-form likelihood:

$$P(\mathbf{A}\,|\,\mathbf{X}) = \frac{P(\mathbf{X}\,|\,\mathbf{A})P(\mathbf{A})}{P(\mathbf{A})}$$

Then we can use any prior for $\mathbf{A}$, but the SBM is a nice option, and allows for more efficient use of data!

$$P(\mathbf{A}\,|\,\mathbf{X}) = \frac{P(\mathbf{X}\,|\,\mathbf{A})P(\mathbf{A}\,|\,\mathbf{b})P(\mathbf{b})}{P(\mathbf{A})}$$

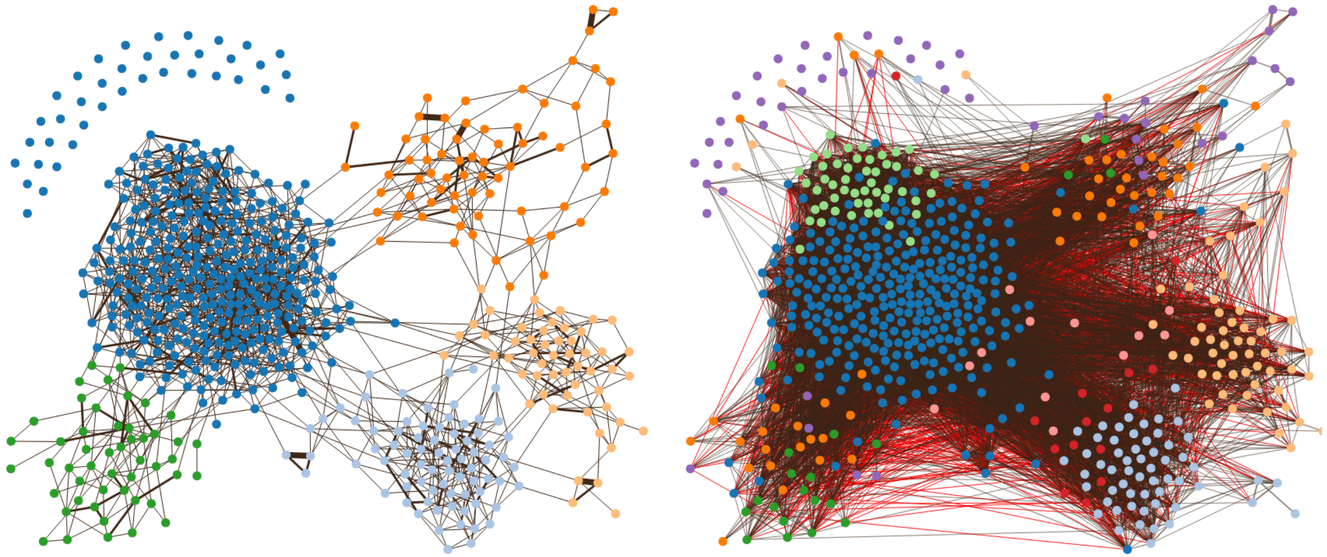
Peixoto. “Network reconstruction and community detection from dynamics,” *PRL (2019)*

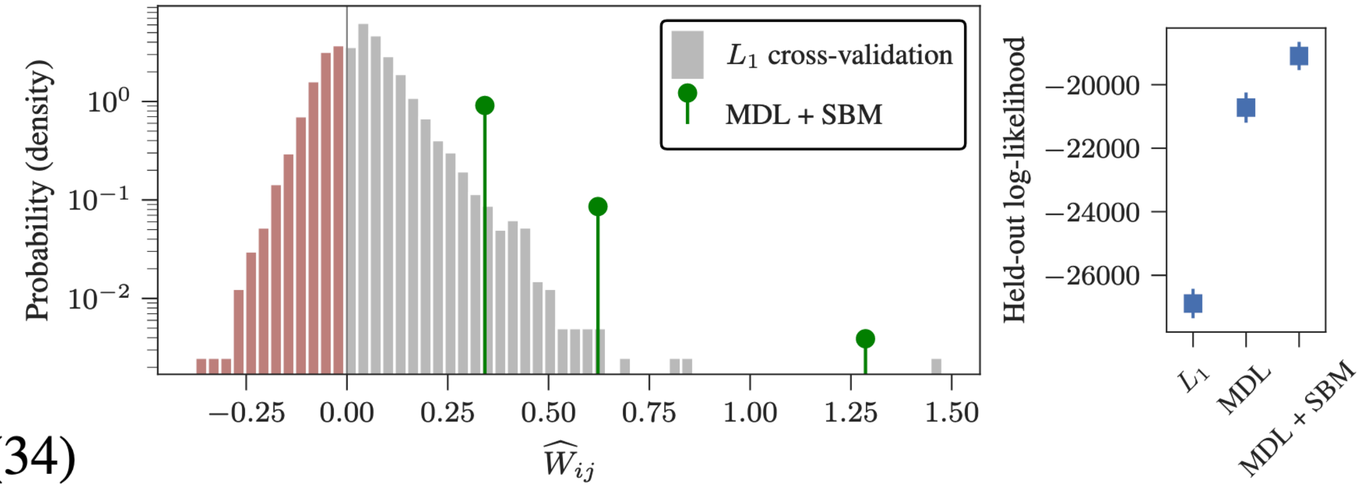
Peixoto. “Network reconstruction via the minimum description length principle,” *PRX (2025)*

Network reconstruction from time series

Benefits of MDL:

- No expensive cross-validation for hyperparameters
  - No to choose a threshold!
- Scalable
- Avoids simultaneous overfitting and underfitting





(34)

Peixoto. "Network reconstruction via the minimum description length principle," *PRX (2025)*

See notebook: `gt-statistical-inference.ipynb`

Tips and recommendations

- Use SBMs as part of your default exploratory pipeline, (e.g. alongside the degree distribution, clustering, assortativity, k-core decomposition, …)
  - It's a very flexible analysis tool, like a "histogram for networks"
- Generally, use all available information in your analysis, i.e.,
  - Include weights if you have them
  - Include layer information if you have it
  - Include bipartite exclusions if you know them, and avoid projecting to unipartite
- When in doubt, use the nested model because of it's improved resolution limit
- Read (and understand) the papers before using these tools in publications
  - Also, ask! Tiago Peixoto, Inverse Complexity Lab, etc. can answer questions

MDL beyond graph-tool

PANINIpy by Alec Kirkley

PANINIpy

latest

Search docs

INSTALLATION AND CONTRIBUTING

Installation Steps

Guidelines for Contributing

METHODS

Binning Temporal Hypergraphs

Clustering Network Populations

Regionalization with Distributional Data

Identifying Network Hubs

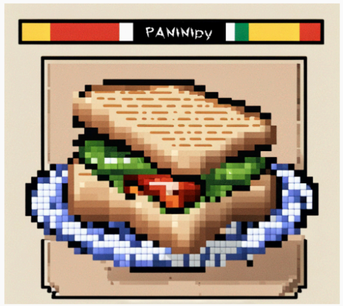
Regionalization with Community Detection

MDL Network Backbones

Partial Rankings

PANINIpy: Package of Algorithms for Nonparametric Inference with Networks in Python

View page source



PANINIpy: Package of Algorithms for Nonparametric Inference with Networks in Python

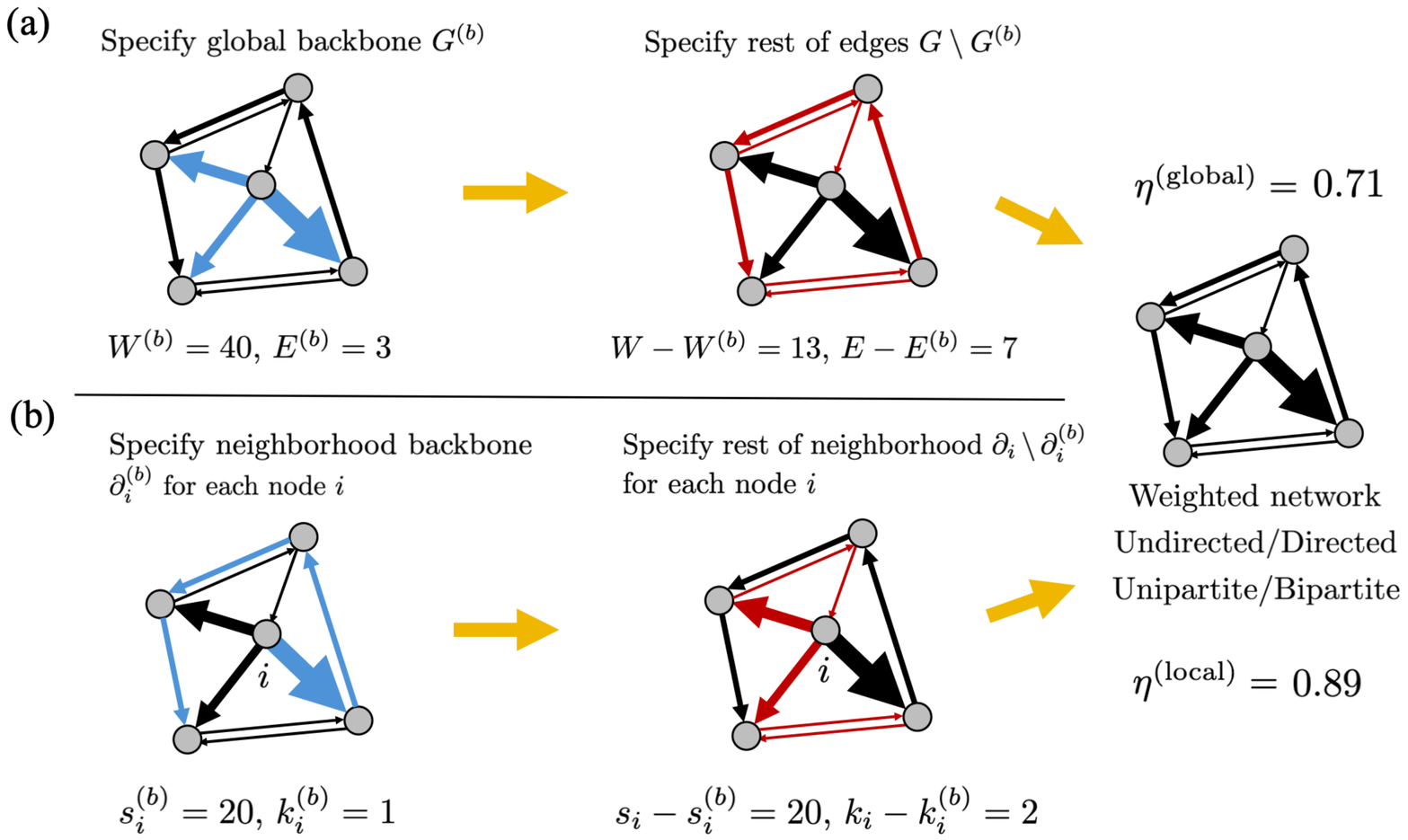
PANINIpy: *PANINIpy* is a Python package providing a collection of nonparametric statistical inference methods for unsupervised learning with network data. Many of these methods are motivated by the Minimum Description Length (MDL) principle and aim to make network inference accessible and robust without the need for parameter tuning. Some benefits of using *PANINIpy* are listed below:

- **Easy Installation**: PANINIpy can be easily installed through pip.
- **Pure Python Implementation**: All modules are written in pure Python, ensuring compatibility with standard Python environments and minimal dependencies.
- **Parameter-Free**: *PANINIpy*'s nonparametric algorithms allow users to extract meaningful summaries of network data without the need to manually adjust free parameters. This reduces potential biases from human aesthetic preferences and implicit algorithmic constraints, and makes the network inference process more user-friendly.
- **Statistical Robustness**: The methods are designed to separate significant structural regularities from noise in network data, making them suitable for real-world datasets that may contain statistical fluctuations and measurement error.

MDL beyond graph-tool

PANINIpy by Alec Kirkley

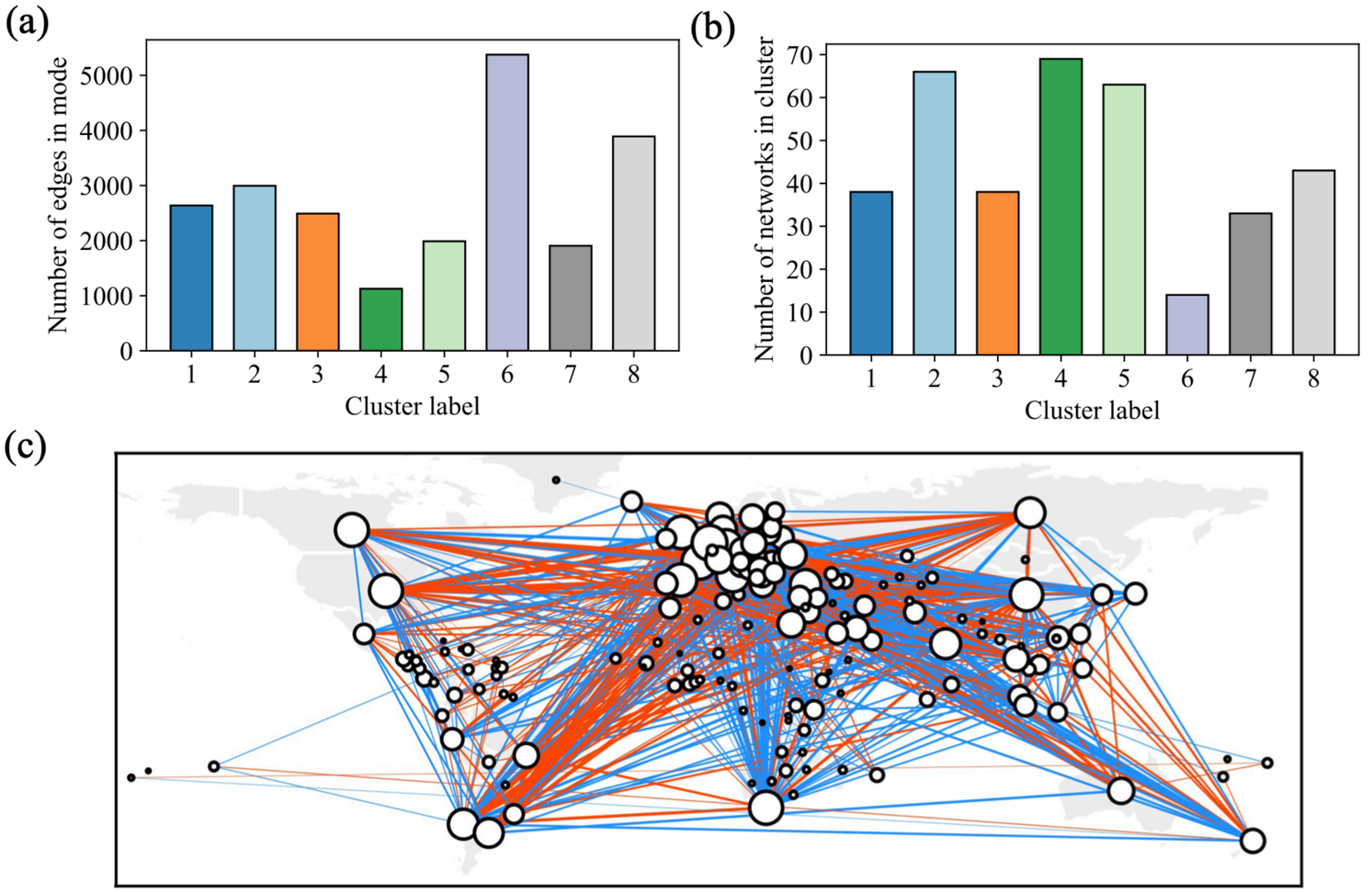
Kirkley. “Fast nonparametric inference of network backbones for weighted graph sparsification” (2025)



MDL beyond graph-tool

PANINIpy by Alec Kirkley

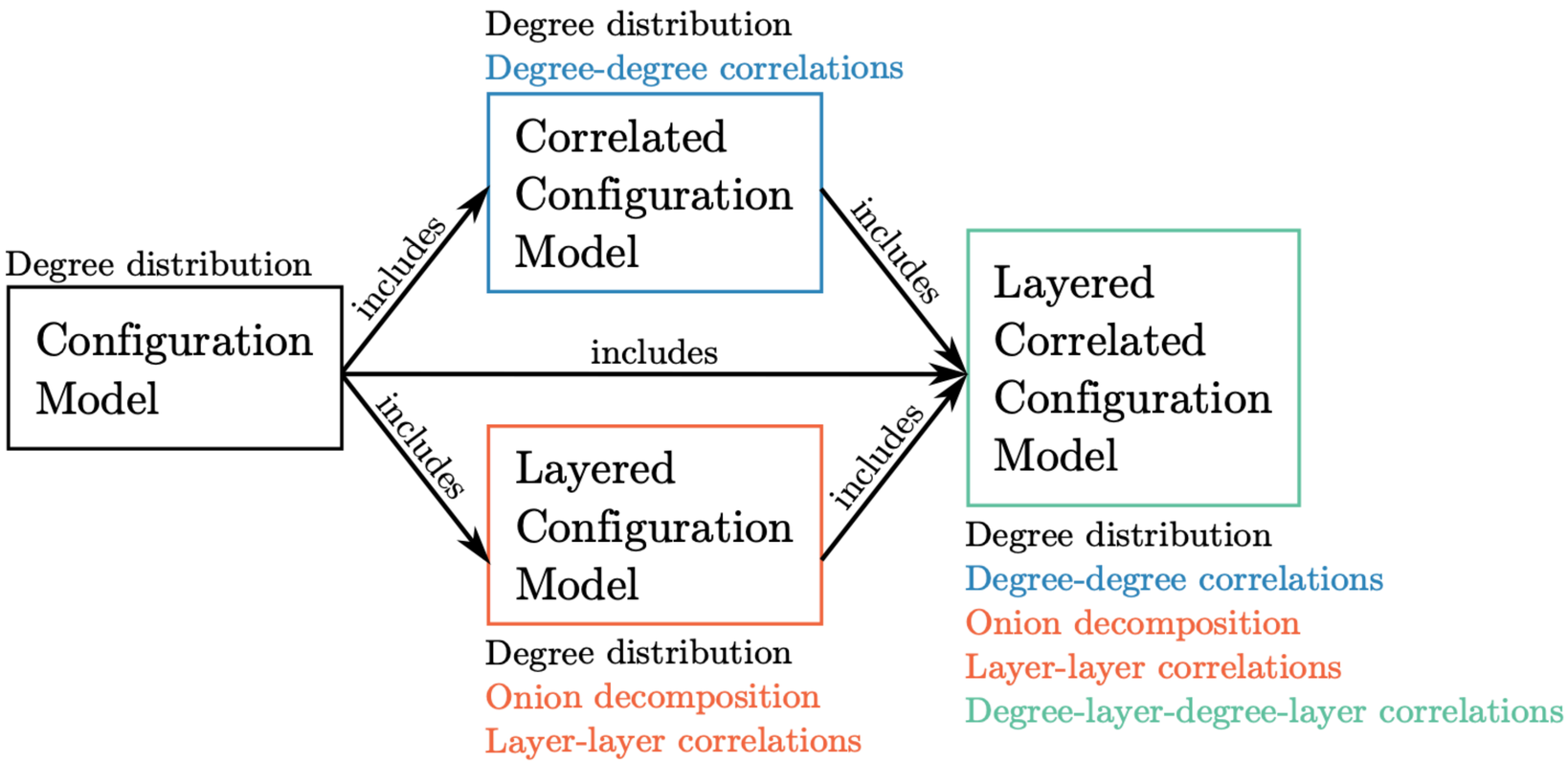
Kirkley et al.
"Compressing network populations with modal networks reveals structural diversity" (2023).



MDL beyond graph-tool

Model selection without Bayesian interpretation

Hébert-Dufresne et al. “Network compression with configuration models and the minimum description length” (2025).



MDL beyond graph-tool

Minimum message length (MML)

Functional reducibility of higher-order networks

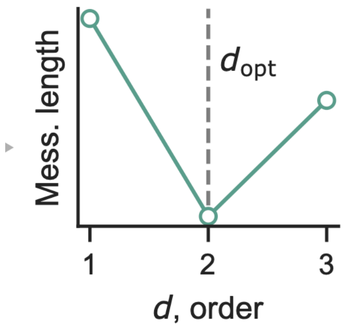
Maxime Lucas,[1,*] Luca Gallo,[2] Arsham Ghavasieh,[3] Federico Battiston,[2,†] and Manlio De Domenico[3,4,5,‡]

[1] *CENTAI Institute, Turin, Italy*
[2] *Department of Network and Data Science, Central European University, 1100 Vienna, Austria*
[3] *Department of Physics and Astronomy "Galileo Galilei", University of Padua, Via F. Marzolo 8, 315126 Padova, Italy*
[4] *Padua Center for Network Medicine, University of Padua, Via F. Marzolo 8, 315126 Padova, Italy*
[5] *Istituto Nazionale di Fisica Nucleare, Sez. Padova, Italy*

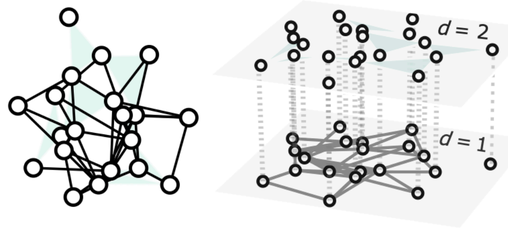
(Dated: May 6, 2024)

Empirical complex systems are widely assumed to be characterized not only by pairwise interactions, but also by higher-order (group) interactions that affect collective phenomena, from metabolic reactions to epidemics. Nevertheless, higher-order networks' superior descriptive power—compared to classical pairwise networks—comes with a much increased model complexity and computational cost. Consequently, it is of paramount importance to establish a quantitative method to determine when such a modeling framework is advantageous with respect to pairwise models, and to which extent it provides a parsimonious description of empirical systems. Here, we propose a principled method, based on information compression, to analyze the reducibility of higher-order networks to lower-order interactions, by identifying redundancies in diffusion processes while preserving the relevant functional information. The analysis of a broad spectrum of empirical systems shows that, although some networks contain non-compressible group interactions, others can be effectively approximated by lower-order interactions—some technological and biological systems even just by pairwise interactions. More generally, our findings mark a significant step towards minimizing the dimensionality of models for complex systems.

Minimize message length



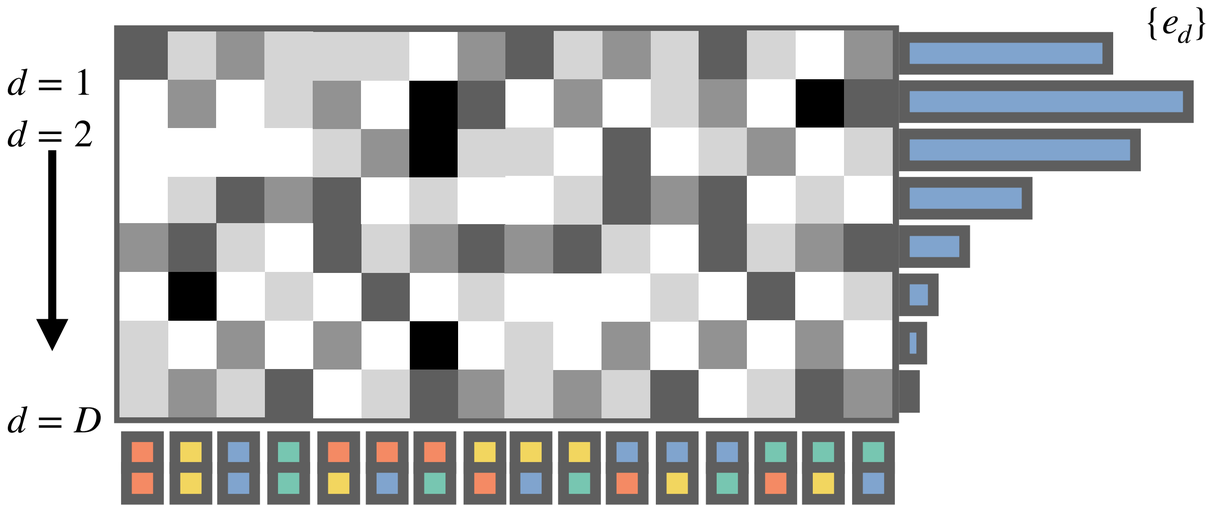
Reduced hypergraph

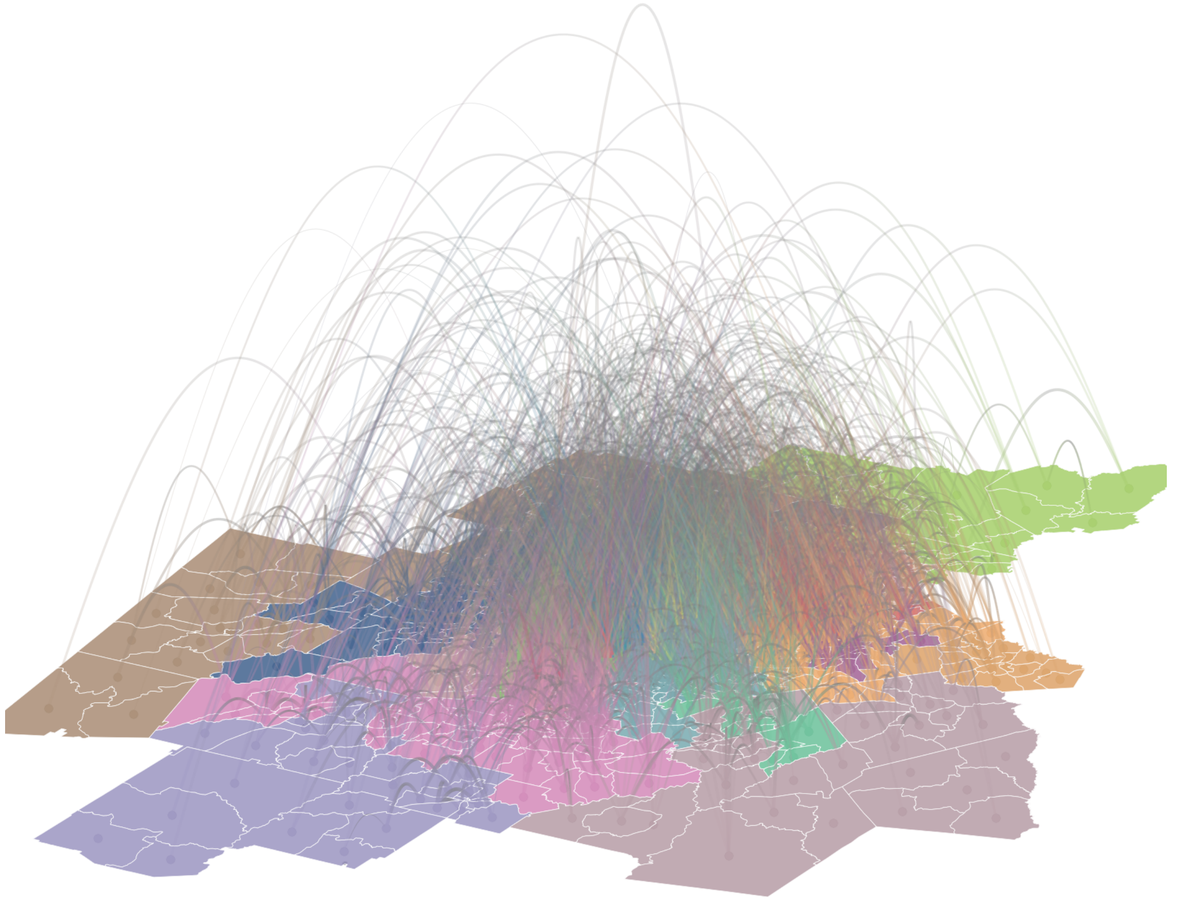


MDL beyond graph-tool

Shameless plug for our ongoing work

Introduce the spatial SBM to perform model selection between geometric and block structures in weighted networks.





Model can be used to perform regionalization by enforcing contiguity constraints.

Joint work with Moritz Laber, Alec Kirkley, and Brennan Klein.

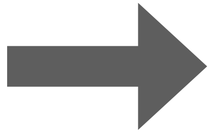
MDL beyond graph-tool

Shameless plug for our ongoing work

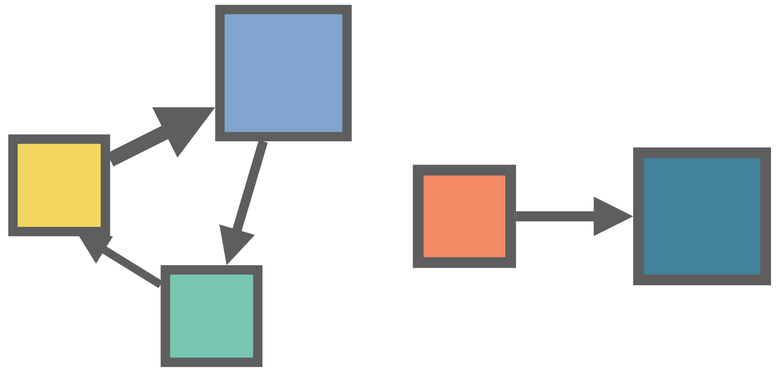
Statistical inference of multiple partitions from network structure, allowing for multiple modes to be detected simultaneously.

Joint work with Lena Mangold, Brennan Klein, and Camille Roth

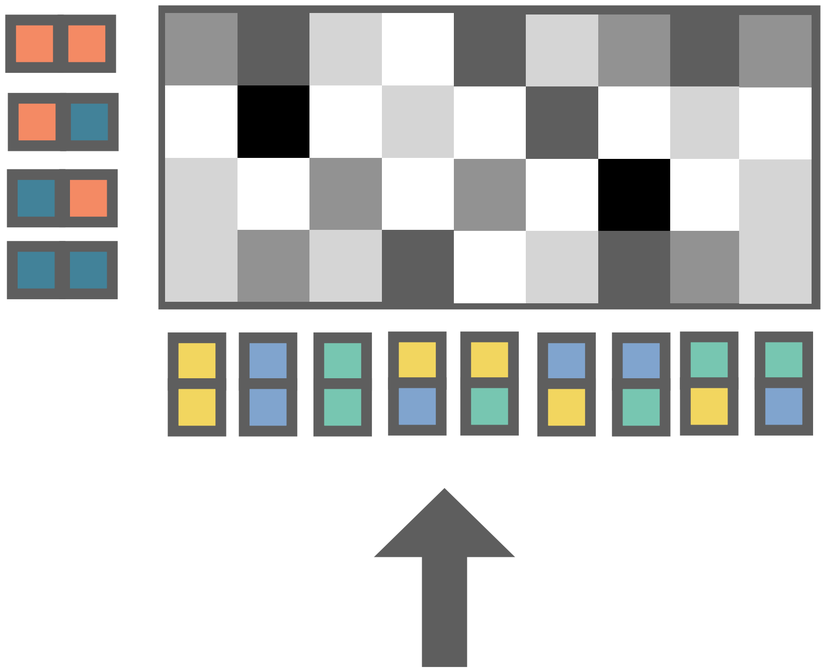
Multiple partitions



Block graphs



Cross-block matrix



References

To get started

Inferential community detection:

- Peixoto. “Bayesian stochastic blockmodeling”, *Advances in Network Clustering and Blockmodeling (2019)*.
- Peixoto. “Descriptive vs. inferential community detection in networks: pitfalls, myths, and half-truths.” *Cambridge University Press (2023)*

Inference and MCMC:

- “Efficient monte carlo and greedy heuristic for the inference of stochastic block model”. *PRE (2014)*
- “Merge-split Markov chain Monte Carlo for community detection,” *PRE (2020)*

References

Models

Nested model: Peixoto 2014

Overlapping groups: Peixoto 2015

Multilayer networks: Peixoto 2015

Bipartite graphs: Gerlach et al. 2018, Hyland et al. 2021

Ranking, hierarchies, and block structure: Peixoto 2022

Network reconstruction and noisy data: Peixoto 2018, Peixoto 2019, Peixoto 2025, Peixoto 2024, Peixoto 2025

Thank you!

Erik Weis

weis.er@northeastern.edu || erikweis.com

Acknowledgements: *Moritz Laber, Lena Mangold, Alec Kirkley, Brennan Klein, Tiago Peixoto, Basti Kusch*

Funding: *Northeastern / NSF PEAR*

Complexity Society Lab

Network Science Institute at Northeastern University