Spectral methods for graph clustering

Ph.D. defence

Lorenzo Dall'Amico

GIPSA lab, Grenoble INP

October 12, 2021

Supervisors:

Romain Couillet

Nicolas Tremblay

Contents

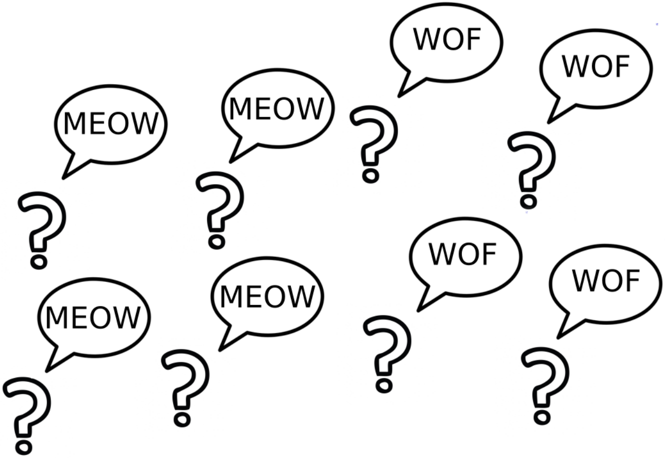
INTRODUCTION

- Graph clustering
- Community detection
- Objectives

Clustering

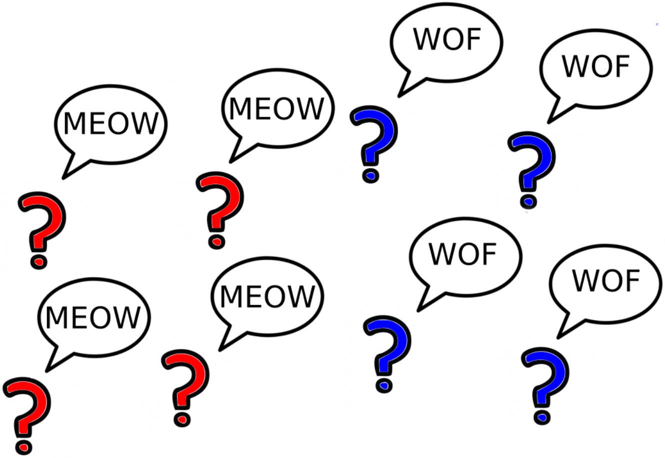
Unsupervised methods for data **categorization**

Unsupervised methods for data **categorization**



Clustering

Unsupervised methods for data **categorization**

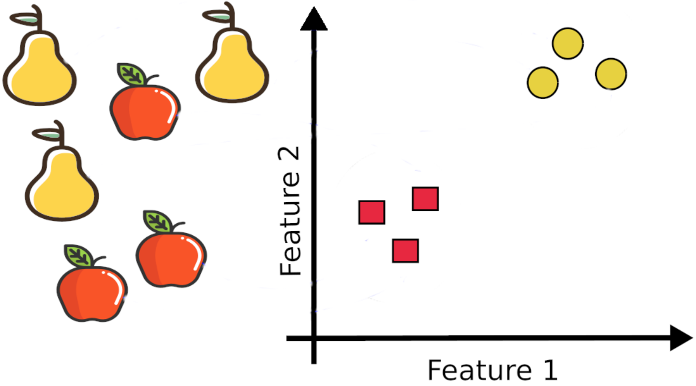


Clustering

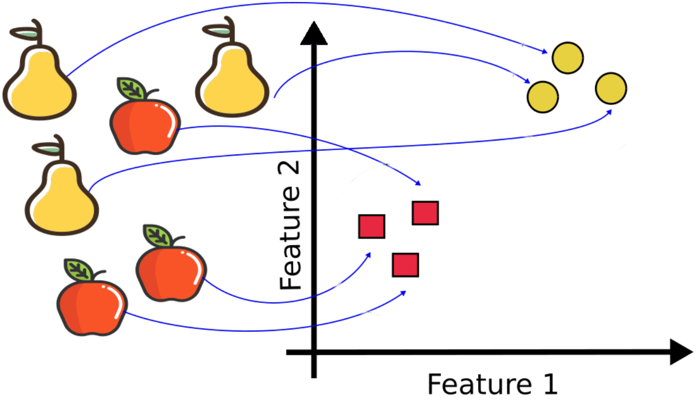
Feature extraction and clustering paradigm

Clustering

Feature extraction and clustering paradigm

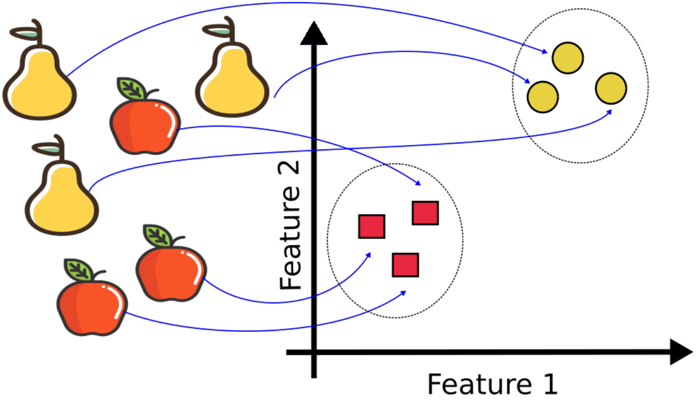


Feature extraction and clustering paradigm



Clustering

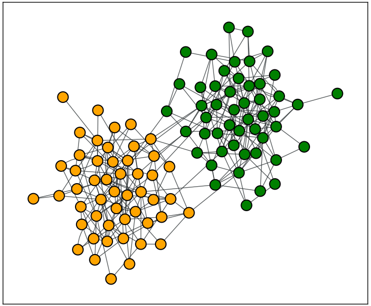
Feature extraction and clustering paradigm

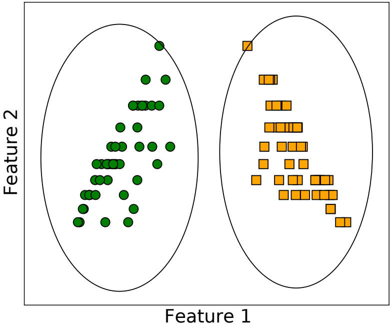


The relevant features define the concept of categories

Graph clustering

A **graph representation** for **pairwise "affinity"**





Graph matrix representation

$\mathcal{G}(\mathcal{V}, \mathcal{E})$, is a graph with $|\mathcal{V}| = n$ nodes.

Definition: adjacency matrix

$$\forall\, 1 \leq i < j \leq n, \quad A_{ij} = \mathbb{1}_{(ij) \in \mathcal{E}}$$

Graph matrix representation

$\mathcal{G}(\mathcal{V}, \mathcal{E})$, is a graph with $|\mathcal{V}| = n$ nodes.

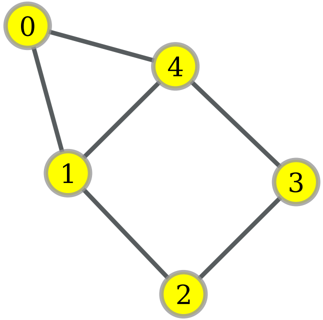
Definition: adjacency matrix

$$\forall\, 1 \leq i < j \leq n, \quad A_{ij} = \mathbb{1}_{(ij) \in \mathcal{E}}$$

Definition: diagonal degree matrix

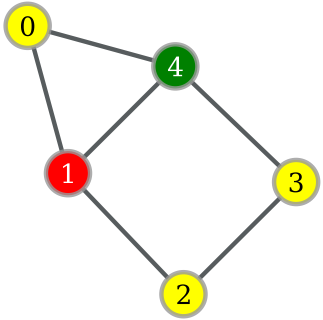
$$\forall\, 1 \leq i \leq j \leq n, \quad D_{ij} = \delta_{ij} \sum_{k \in \mathcal{V}} A_{ij}$$

Graph matrix representation



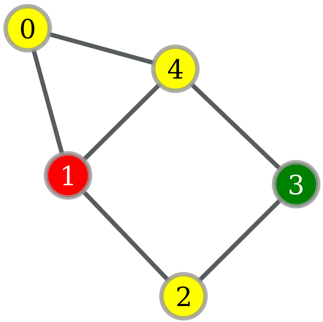
$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Graph matrix representation



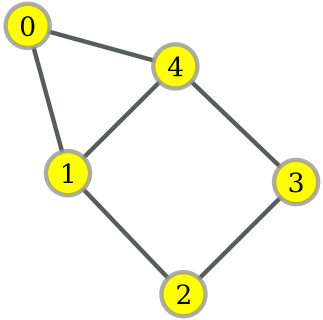
$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Graph matrix representation



$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Graph matrix representation



$$D = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{pmatrix}$$

Spectral clustering

Graph $\rightarrow$ Matrix $\rightarrow$ Eigenvectors embedding

Spectral clustering

Graph $\to$ Matrix $\to$ Eigenvectors embedding

Typical spectral clustering algorithm

Input: $\mathcal{G}(\mathcal{V}, \mathcal{E}), k$

- $M \in \mathbb{R}^{n \times n}$: graph matrix representation
- k eigenvectors of M in the columns of $X \in \mathbb{R}^{n \times k}$
- Clustering in k-dimensions (k-means)

Output: $\ell \in \{1, \ldots, k\}^n$, labelling vector

3 years in 40 minutes...

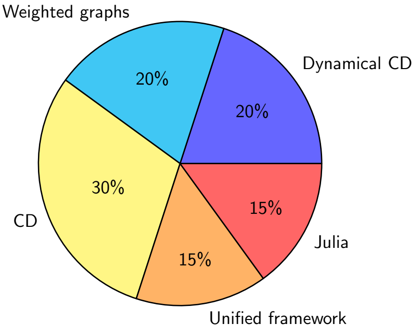


Figure: My PhD

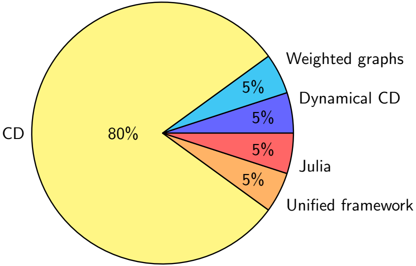


Figure: My presentation

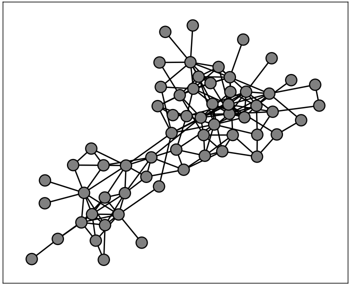
Community detection: problem position

Problem: given $\mathcal{G}(\mathcal{V}, \mathcal{E})$, find a partition of $\mathcal{V}$ to recover the community labels

Community detection: problem position

Problem: given $\mathcal{G}(\mathcal{V}, \mathcal{E})$, find a partition of $\mathcal{V}$ to recover the community labels

INPUT



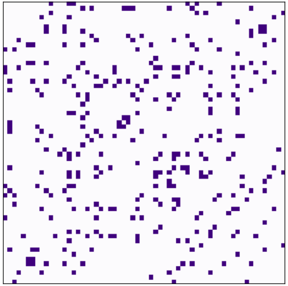
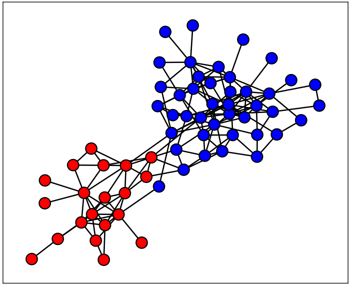


Figure: The dolphin network (Lusseau 2003)

Community detection: problem position

Problem: given $\mathcal{G}(\mathcal{V}, \mathcal{E})$, find a partition of $\mathcal{V}$ to recover the community labels

OUTPUT



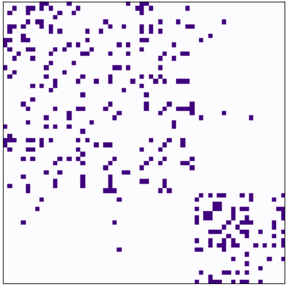


Figure: The dolphin network (Lusseau 2003)

Spectral clustering for community detection

Recall: compute k eigenvectors of M to embed the nodes in $\mathbb{R}^k$

Spectral clustering for community detection

Recall: compute k eigenvectors of M to embed the nodes in $\mathbb{R}^k$

Popular choices for M in community detection are

- A, adjacency matrix
- $D - A$, graph Laplacian matrix
- $D^{-1}A, D^{-1/2}AD^{-1/2}$, normalized Laplacian matrices

Spectral clustering for community detection

Recall: compute k eigenvectors of M to embed the nodes in $\mathbb{R}^k$

Popular choices for M in community detection are

- A, adjacency matrix
- $D - A$, graph Laplacian matrix
- $D^{-1}A, D^{-1/2}AD^{-1/2}$, normalized Laplacian matrices

Poor performances on **sparse** and **heterogeneous** graphs

Sparsity

Real graphs are typically **sparse**

Dataset	Size	Average degree
Dolphins	62	5
Polbooks	105	8,4
Football	115	10,7
Polblogs	1.222	27,4
Facebook	4.039	43,7
GNutella P2P	6.301	6,6
Astrophysics	18.775	21,1
Condensed matter	23.133	8,1
Email Enron	36.692	10

Source : Stanford Large Network Dataset Collection

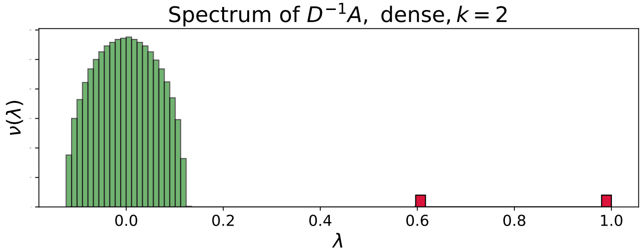
Sparsity

Real graphs are typically **sparse**

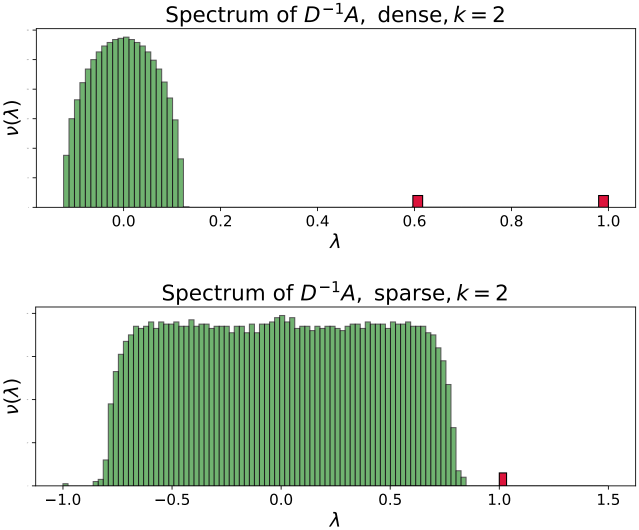
Dataset	Size	Average degree
Dolphins	62	5
Polbooks	105	8,4
Football	115	10,7
Polblogs	1.222	27,4
Facebook	4.039	43,7
GNutella P2P	6.301	6,6
Astrophysics	18.775	21,1
Condensed matter	23.133	8,1
Email Enron	36.692	10

Source : Stanford Large Network Dataset Collection

Spectral clustering in sparse graphs



Spectral clustering in sparse graphs



Real graphs typically have a **broad degree distribution**

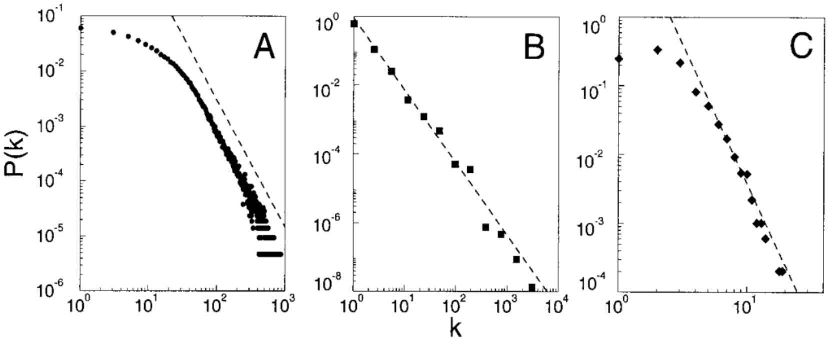


Figure: Degree distribution for various networks. Source: Barabasi, *Emergence of Scaling in Random Networks*, Science 1999

Spectral clustering in heterogeneous graphs

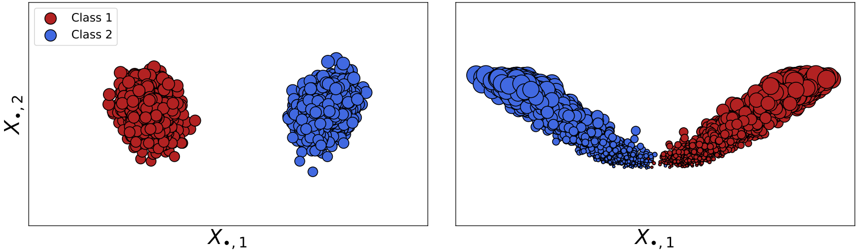


Figure: Spectral node embedding. **Left**: homogeneous degree distribution. **Right**: power law degree distribution

The Degree-corrected stochastic block model (DCSBM)

The Degree-corrected stochastic block model (DCSBM)

$$\mathbb{P}(A_{ij} = 1) = \theta_i \theta_j \cdot \frac{C_{\ell_i \ell_j}}{n}$$

The Degree-corrected stochastic block model (DCSBM)

$$\mathbb{P}(A_{ij} = 1) = \theta_i \theta_j \cdot \frac{C_{\ell_i \ell_j}}{n}$$

- $\boldsymbol{\theta} \in \mathbb{R}^n$: intrinsic "probability" of connection
  - $\frac{1}{n} \sum_{i \in \mathcal{V}} \theta_i = 1$
  - $\frac{1}{n} \sum_{i \in \mathcal{V}} \theta_i^2 = \Phi = O_n(1)$: broadness of the degree distribution

The Degree-corrected stochastic block model (DCSBM)

$$\mathbb{P}(A_{ij} = 1) = \theta_i \theta_j \cdot \frac{C_{\ell_i \ell_j}}{n}$$

- $\boldsymbol{\theta} \in \mathbb{R}^n$: intrinsic "probability" of connection
  - $\frac{1}{n}\sum_{i \in \mathcal{V}} \theta_i = 1$
  - $\frac{1}{n}\sum_{i \in \mathcal{V}} \theta_i^2 = \Phi = O_n(1)$: broadness of the degree distribution
- $C \in R^{k \times k}$: class affinity matrix
- $\ell \in \{1, \ldots, k\}^n$: label vector

The Degree-corrected stochastic block model (DCSBM)

$$\mathbb{P}(A_{ij} = 1) = \theta_i \theta_j \cdot \frac{C_{\ell_i \ell_j}}{n}$$

- $\boldsymbol{\theta} \in \mathbb{R}^n$: intrinsic "probability" of connection
  - $\frac{1}{n} \sum_{i \in \mathcal{V}} \theta_i = 1$
  - $\frac{1}{n} \sum_{i \in \mathcal{V}} \theta_i^2 = \Phi = O_n(1)$: broadness of the degree distribution
- $C \in R^{k \times k}$: class affinity matrix
- $\ell \in \{1, \ldots, k\}^n$: label vector
- $\mathbb{E}[d_i] = O_n(1)$

The Degree-corrected stochastic block model (DCSBM)

$$\mathbb{P}(A_{ij} = 1) = \theta_i \theta_j \cdot \frac{C_{\ell_i \ell_j}}{n}$$

- $\boldsymbol{\theta} \in \mathbb{R}^n$: intrinsic "probability" of connection
  - $\frac{1}{n} \sum_{i \in \mathcal{V}} \theta_i = 1$
  - $\frac{1}{n} \sum_{i \in \mathcal{V}} \theta_i^2 = \Phi = O_n(1)$: broadness of the degree distribution
- $C \in R^{k \times k}$: class affinity matrix
- $\ell \in \{1, \ldots, k\}^n$: label vector
- $\mathbb{E}[d_i] = O_n(1)$

Objective: infer ℓ from A

Detectability threshold

Detectability threshold

- k classes of equal size
- $C = \begin{pmatrix} c_{\text{in}} & c_{\text{out}} & c_{\text{out}} & \cdots \\ c_{\text{out}} & c_{\text{in}} & c_{\text{out}} & \cdots \\ \vdots & \vdots & \ddots & \vdots \\ \cdots & c_{\text{out}} & c_{\text{out}} & c_{\text{in}} \end{pmatrix}$

Detectability threshold

- k classes of equal size
- $C = \begin{pmatrix} c_{\text{in}} & c_{\text{out}} & c_{\text{out}} & \cdots \\ c_{\text{out}} & c_{\text{in}} & c_{\text{out}} & \cdots \\ \vdots & \vdots & \ddots & \vdots \\ \cdots & c_{\text{out}} & c_{\text{out}} & c_{\text{in}} \end{pmatrix}$

Detection in polynomial time if

$$\alpha = f(c_{\text{in}}, c_{\text{out}}, \Phi, k) > \alpha_{\text{c}}$$

Detectability threshold

- k classes of equal size
- $C = \begin{pmatrix} c_{\rm in} & c_{\rm out} & c_{\rm out} & \dots \\ c_{\rm out} & c_{\rm in} & c_{\rm out} & \dots \\ \vdots & \vdots & \ddots & \vdots \\ \dots & c_{\rm out} & c_{\rm out} & c_{\rm in} \end{pmatrix}$

Detection in polynomial time if

$$\alpha = f(c_{\rm in}, c_{\rm out}, \Phi, k) > \alpha_{\rm c}$$

Typical choices of M do not achieve the $\alpha_{\rm c}$ threshold

Objectives

Define a new spectral clustering algorithm

Objectives

Define a new spectral clustering algorithm

- Inspired by the DCSBM but applicable to **any** input graph

Objectives

Define a new spectral clustering algorithm

- Inspired by the DCSBM but applicable to **any** input graph
- Dealing with **sparsity** and **degree heterogeneity**

Objectives

Define a new spectral clustering algorithm

- Inspired by the DCSBM but applicable to **any** input graph
- Dealing with **sparsity** and **degree heterogeneity**
- Reaching the **detectability threshold** on sparse DCSBM graphs

Contents

PHYSICS INSPIRED METHODS

- The Bethe-Hessian
- The non-backtracking
- Challenges

The Bethe-Hessian

Ising Hamiltonian on $\mathcal{G}$ for $\boldsymbol{s} \in \{-1, 1\}^n$

$$\mathcal{H}(\boldsymbol{s}) = -\boldsymbol{s}^T A \boldsymbol{s} = -\sum_{(ij) \in \mathcal{E}} s_i s_j$$

The Bethe-Hessian

Ising Hamiltonian on $\mathcal{G}$ for $\boldsymbol{s} \in \{-1, 1\}^n$

$$\mathcal{H}(\boldsymbol{s}) = -\boldsymbol{s}^T A \boldsymbol{s} = -\sum_{(ij)\in\mathcal{E}} s_i s_j$$

Low energetic configurations with all spins in the same class are aligned

The Bethe-Hessian

Ising Hamiltonian on $\mathcal{G}$ for $\boldsymbol{s} \in \{-1,1\}^n$

$$\mathcal{H}(\boldsymbol{s}) = -\boldsymbol{s}^T A \boldsymbol{s} = -\sum_{(ij)\in\mathcal{E}} s_i s_j$$

Low energetic configurations with all spins in the same class are aligned

Free energy for $\boldsymbol{m} \in \mathbb{R}^n$

$$F(\boldsymbol{m}) = \underbrace{U(\boldsymbol{m})}_{\text{internal energy}} - T \underbrace{S(\boldsymbol{m})}_{\text{entropy}}$$

The minima of $F(\boldsymbol{m})$ are informative

The Bethe-Hessian

- Ising Hamiltonian $\mathcal{H}(\boldsymbol{s})$

The Bethe-Hessian

- Ising Hamiltonian $\mathcal{H}(\boldsymbol{s})$
- Temperature-like parameter r

The Bethe-Hessian

- Ising Hamiltonian $\mathcal{H}(\boldsymbol{s})$
- Temperature-like parameter r
- Bethe approximation

The Bethe-Hessian

- Ising Hamiltonian $\mathcal{H}(\boldsymbol{s})$
- Temperature-like parameter r
- Bethe approximation
- Hessian of the Bethe free energy

The Bethe-Hessian

- Ising Hamiltonian $\mathcal{H}(\boldsymbol{s})$
- Temperature-like parameter r
- Bethe approximation
- Hessian of the Bethe free energy

Definition: Bethe-Hessian family of matrices

$$H_r = (r^2 - 1)I_n + D - rA, \quad r \geq 1$$

The Bethe-Hessian

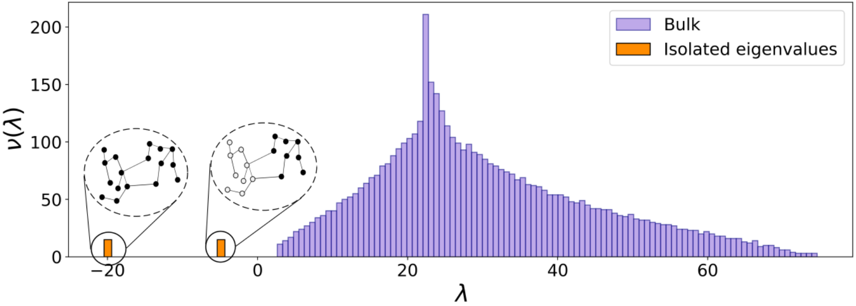


Figure: Spectrum of H_r for r fixed and $k = 2$

The non-backtracking

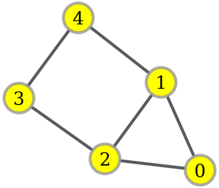
Definition: non-backtracking matrix

$$\forall\ (ij),(kl) \in \mathcal{E}_{\rm d}, \quad B_{(ij),(kl)} = \delta_{jk}(1-\delta_{il})$$

The non-backtracking

Definition: non-backtracking matrix

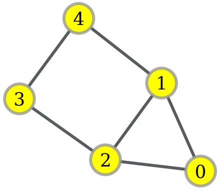
$$\forall\ (ij),(kl) \in \mathcal{E}_{\mathrm{d}}, \quad B_{(ij),(kl)} = \delta_{jk}(1-\delta_{il})$$

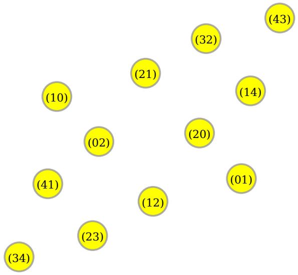


The non-backtracking

Definition: non-backtracking matrix

$$\forall\ (ij),(kl) \in \mathcal{E}_\mathrm{d}, \quad B_{(ij),(kl)} = \delta_{jk}(1-\delta_{il})$$

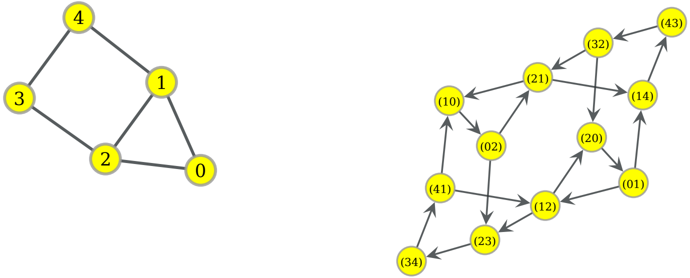




The non-backtracking

DEFINITION: non-backtracking matrix

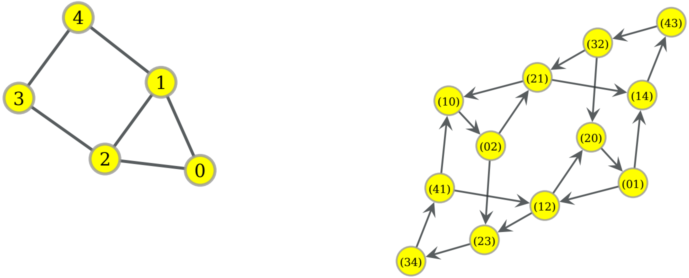
$$\forall\ (ij),(kl) \in \mathcal{E}_{\mathrm{d}}, \quad B_{(ij),(kl)} = \delta_{jk}(1-\delta_{il})$$



The non-backtracking

DEFINITION: non-backtracking matrix

$$\forall\ (ij),(kl) \in \mathcal{E}_\mathrm{d}, \quad B_{(ij),(kl)} = \delta_{jk}(1-\delta_{il})$$



$$\det(B - rI_{|\mathcal{E}_\mathrm{d}|}) \propto \det(H_r)$$

Spectrum of B (DCSBM)

- For $\alpha > \alpha_c$, k real isolated eigenvalues
- All complex eigenvalues satisfy $|\lambda_i(B)| \leq \sqrt{\rho(B)} + o_n(1)$

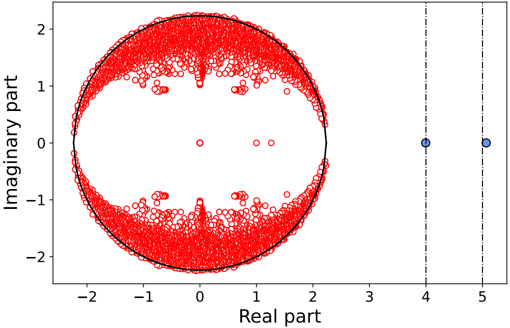


Figure: Spectrum of B for $k = 2$, sparse DCSBM

Spectrum of H_r (DCSBM)

For $r = \sqrt{\rho(B)}$ (Saade 2014)

- If $\alpha > \alpha_c$, k isolated negative eigenvalues
- All bulk eigenvalues are positive

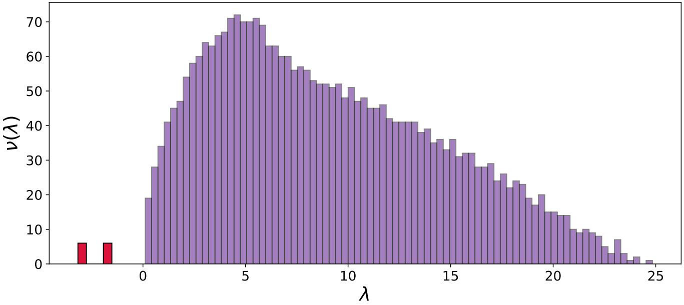


Figure: Spectrum of $H_{\sqrt{\rho(B)}}$ for $k = 2$, sparse DCSBM

Challenges

Both $H_{\sqrt{\rho(B)}}$ and B

✔ achieve **detectability** when $\alpha > \alpha_{\rm c}$

Challenges

Both $H_{\sqrt{\rho(B)}}$ and B

- ✔ achieve **detectability** when $\alpha > \alpha_c$
- ✘ do not deal with **heterogeneity**

Challenges

Both $H_{\sqrt{\rho(B)}}$ and B

✔ achieve **detectability** when $\alpha > \alpha_c$

✘ do not deal with **heterogeneity**

Q1 : deal with **sparsity** and **heterogeneity** at once

Challenges

Both $H_{\sqrt{\rho(B)}}$ and B

✔ achieve **detectability** when $\alpha > \alpha_c$

✘ do not deal with **heterogeneity**

Q1 : deal with **sparsity** and **heterogeneity** at once

Q2 : find the optimal $r \neq \sqrt{\rho(B)}$ for H_r

Challenges

Both $H_{\sqrt{\rho(B)}}$ and B

✔ achieve **detectability** when $\alpha > \alpha_c$

✘ do not deal with **heterogeneity**

Q1 : deal with **sparsity** and **heterogeneity** at once

Q2 : find the optimal $r \neq \sqrt{\rho(B)}$ for H_r

Q3 : efficiently estimate the optimal r

Contents

MAIN CONTRIBUTION

- Optimal Bethe-Hessian
- DCSBM
- Algorithm
- Performance
- A unified framework

The definition of ζ_p

Contribution: definition of optimal value of r

$$H_r = (r^2 - 1)I_n + D - rA$$

The definition of ζ_p

Contribution: definition of optimal value of r

$$H_r = (r^2 - 1)I_n + D - rA$$

Let $\lambda_p^{\uparrow}(M)$ be the p-th smallest eigenvalue of M

The definition of ζ_p

Contribution: definition of optimal value of r

$$H_r = (r^2 - 1)I_n + D - rA$$

Let $\lambda_p^{\uparrow}(M)$ be the p-th smallest eigenvalue of M

Definition: ζ_p

For each connected component of $\mathcal{G}$

$$\zeta_p = \min_{r \geq 1}\{r \; : \; \lambda_p^{\uparrow}(H_r) = 0\}$$

The definition of ζ_p

Contribution: definition of optimal value of r

$$H_r = (r^2 - 1)I_n + D - rA$$

Let $\lambda_p^{\uparrow}(M)$ be the p-th smallest eigenvalue of M

DEFINITION: ζ_p

For each connected component of $\mathcal{G}$

$$\zeta_p = \min_{r \geq 1}\{r \;:\; \lambda_p^{\uparrow}(H_r) = 0\}$$

- Well defined for any $\mathcal{G}$

The definition of ζ_p

Contribution: definition of optimal value of r

$$H_r = (r^2 - 1)I_n + D - rA$$

Let $\lambda_p^{\uparrow}(M)$ be the p-th smallest eigenvalue of M

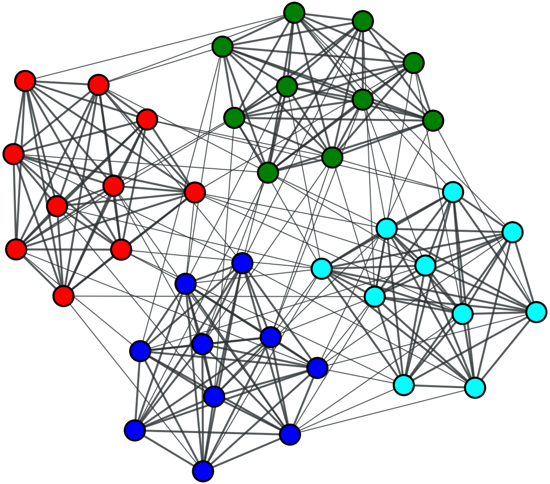
Definition: ζ_p

For each connected component of $\mathcal{G}$

$$\zeta_p = \min_{r \geq 1}\{r \ : \ \lambda_p^{\uparrow}(H_r) = 0\}$$

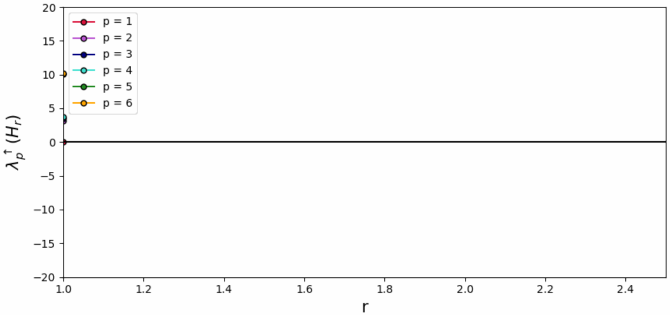
- Well defined for any $\mathcal{G}$
- A "property" of the graph

The definition of ζ_p



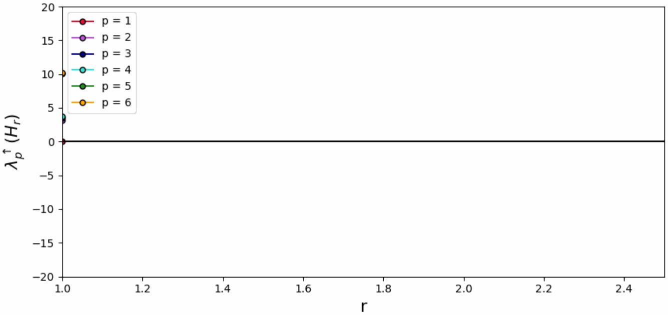
The definition of ζ_p

$$H_r = (r^2 - 1)I_n + D - rA$$



The definition of ζ_p

$$H_r = (r^2 - 1)I_n + D - rA$$



The definition of ζ_p

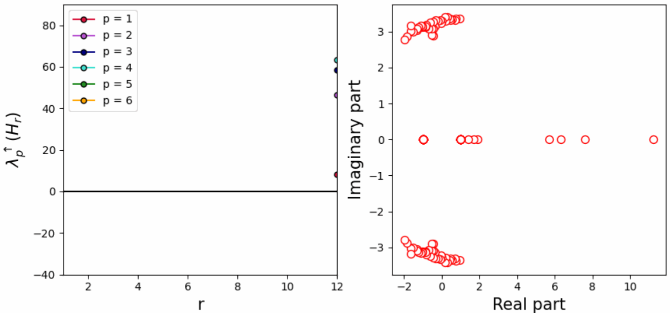
$$\det(B - rI_{|\mathcal{E}_\mathrm{d}|}) \propto \det(H_r)$$

The ζ_p's are eigenvalues of B

The definition of ζ_p

$$\det(B - rI_{|\mathcal{E}_d|}) \propto \det(H_r)$$

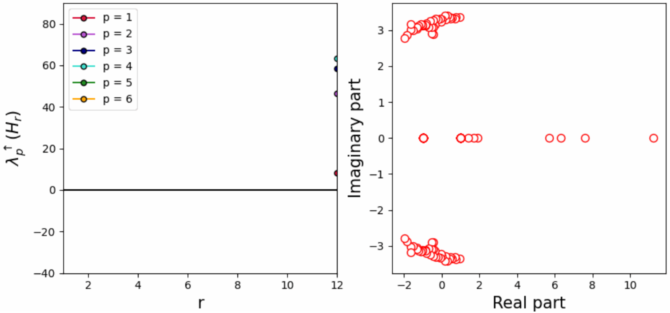
The ζ_p's are eigenvalues of B



The definition of ζ_p

$$\det(B - rI_{|\mathcal{E}_\mathrm{d}|}) \propto \det(H_r)$$

The ζ_p's are eigenvalues of B



DCSBM: Properties of ζ_p

DCSBM: Properties of ζ_p

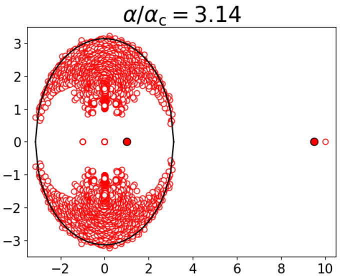
- The largest p for which ζ_p is defined is k (**Estimate of** k)

DCSBM: Properties of ζ_p

- The largest p for which ζ_p is defined is k (**Estimate of** k)
- $\zeta_p = \frac{\rho(B)}{\lambda_p^{\downarrow|\cdot|}(B)} + o_n(1)$

DCSBM: Properties of ζ_p

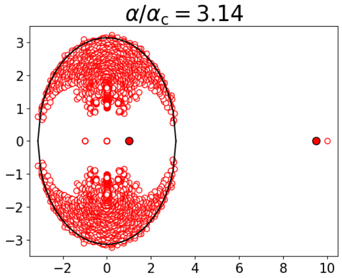
- The largest p for which ζ_p is defined is k (**Estimate of** k)
- $\zeta_p = \frac{\rho(B)}{\lambda_p^{\downarrow|\cdot|}(B)} + o_n(1)$



- hard problems = Large ζ_p
  = high temperature
- ζ_p:detection when $\alpha > \alpha_c$

Properties of ζ_p

- The largest p for which ζ_p is defined is k (**Estimate of** k)
- $\zeta_p = \frac{\rho(B)}{\lambda_p^{\downarrow|\cdot|}(B)} + o_n(1)$



- hard problems = Large ζ_p = high temperature
- ζ_p:detection when $\alpha > \alpha_c$

Properties of x_p

$$H_{\zeta_p}\boldsymbol{x}_p = [(\zeta_p^2 - 1)I_n + D - \zeta_p A]\boldsymbol{x}_p = 0$$

Properties of x_p

$$H_{\zeta_p} \boldsymbol{x}_p = [(\zeta_p^2 - 1) I_n + D - \zeta_p A] \boldsymbol{x}_p = 0$$

For $k = 2$ let $\sigma_i \in \{\pm 1\}$ be the label

Properties of $\boldsymbol{x}_p$

$$H_{\zeta_p}\boldsymbol{x}_p = [(\zeta_p^2 - 1)I_n + D - \zeta_p A]\boldsymbol{x}_p = 0$$

For $k = 2$ let $\sigma_i \in \{\pm 1\}$ be the label

The eigenvector $\boldsymbol{x}_2 \approx \boldsymbol{\sigma}$ for all D

Properties of $\boldsymbol{x}_p$

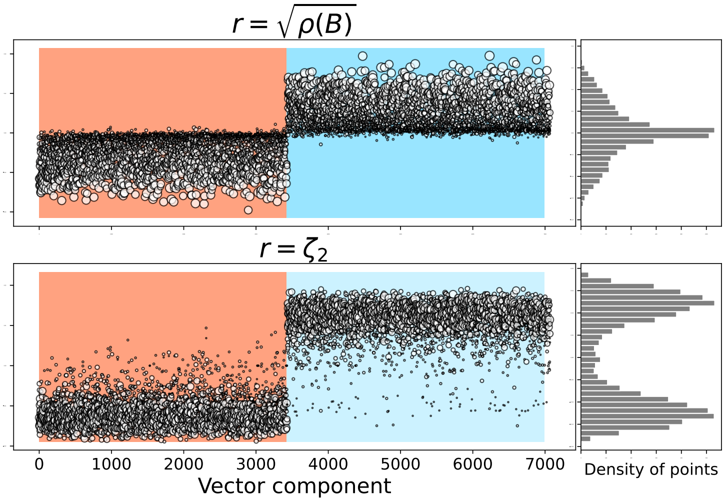
$$H_{\zeta_p}\boldsymbol{x}_p = [(\zeta_p^2 - 1)I_n + D - \zeta_p A]\boldsymbol{x}_p = 0$$

For $k = 2$ let $\sigma_i \in \{\pm 1\}$ be the label

The eigenvector $\boldsymbol{x}_2 \approx \boldsymbol{\sigma}$ for all D

Valid also for $k > 2$

Properties of x_p



Proposed algorithm

CD with the optimal BH

Input: connected graph $\mathcal{G}$, k

Proposed algorithm

CD with the optimal BH

Input: connected graph $\mathcal{G}$, k

For $p = 1 : k$

- $\zeta_p \ : \ \lambda_p^{\uparrow}(H_{\zeta_p}) = 0$

Proposed algorithm

CD with the optimal BH

Input: connected graph $\mathcal{G}$, k

For $p = 1 : k$

- $\zeta_p \ : \ \lambda_p^{\uparrow}(H_{\zeta_p}) = 0$
- $X_{\bullet,p} \leftarrow \boldsymbol{x}_p \ : \ H_{\zeta_p}\boldsymbol{x}_p = 0$

Proposed algorithm

CD with the optimal BH

Input: connected graph $\mathcal{G}$, k

For $p = 1 : k$

- $\zeta_p \ : \ \lambda_p^{\uparrow}(H_{\zeta_p}) = 0$
- $X_{\bullet,p} \leftarrow \boldsymbol{x}_p \ : \ H_{\zeta_p}\boldsymbol{x}_p = 0$

- Clustering in k-dimensions

Return: $\ell \in \{1, \ldots, k\}^n$, label assigment

Proposed algorithm

CD with the optimal BH

Input: connected graph $\mathcal{G}$, $k \leftarrow \left|\{i \; : \; \lambda_i(H_{\sqrt{\rho(B)}}) < 0\}\right|$

For $p = 1 : k$

- $\zeta_p \; : \; \lambda_p^{\uparrow}(H_{\zeta_p}) = 0$
- $X_{\bullet,p} \leftarrow \boldsymbol{x}_p \; : \; H_{\zeta_p}\boldsymbol{x}_p = 0$

- Clustering in k-dimensions

Return: $\ell \in \{1, \dots, k\}^n$, label assigment

Proposed algorithm

CD with the optimal BH

Input: connected graph $\mathcal{G}$, $k \leftarrow \left|\{i \; : \; \lambda_i(H_{\sqrt{\rho(B)}}) < 0\}\right|$

For $p = 1 : k$

- $\zeta_p \; : \; \lambda_p^{\uparrow}(H_{\zeta_p}) = 0$
- $X_{\bullet,p} \leftarrow \boldsymbol{x}_p \; : \; H_{\zeta_p}\boldsymbol{x}_p = 0$

- Clustering in k-dimensions

Random projection
+
polynomial approximation

Return: $\ell \in \{1, \ldots, k\}^n$, label assigment

Proposed algorithm

CD with the optimal BH

Input: connected graph $\mathcal{G}$, $k \leftarrow \left|\{i \; : \; \lambda_i(H_{\sqrt{\rho(B)}}) < 0\}\right|$

For $p = 1 : k$

- $\zeta_p \; : \; \lambda_p^{\uparrow}(H_{\zeta_p}) = 0$
- $X_{\bullet,p} \leftarrow \boldsymbol{x}_p \; : \; H_{\zeta_p}\boldsymbol{x}_p = 0$

- Clustering in k-dimensions

Return: $\ell \in \{1, \ldots, k\}^n$, label assigment

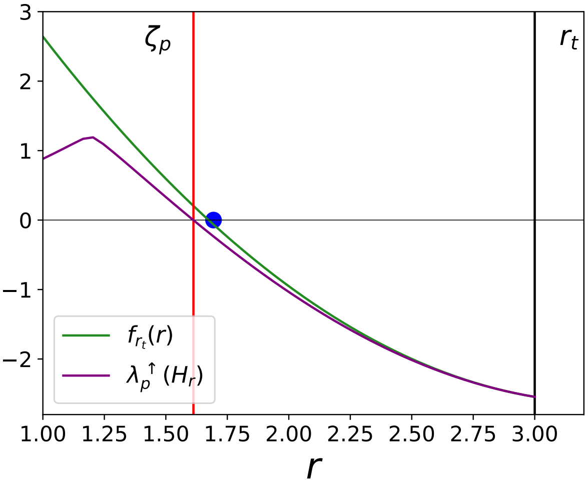
Random projection
+
polynomial approximation

Efficient estimate of ζ_p

$\zeta_p \leftarrow r \; : \; \lambda_p^{\uparrow}(H_r) = 0$: grid search is **inefficient**

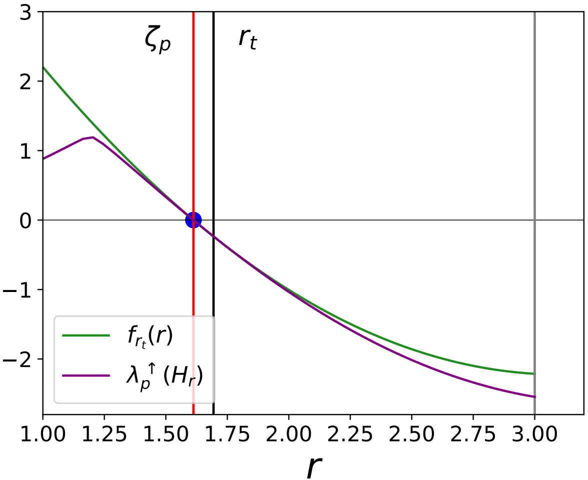
Efficient estimate of ζ_p

$\zeta_p \leftarrow r \;:\; \lambda_p^{\uparrow}(H_r) = 0$: grid search is **inefficient**



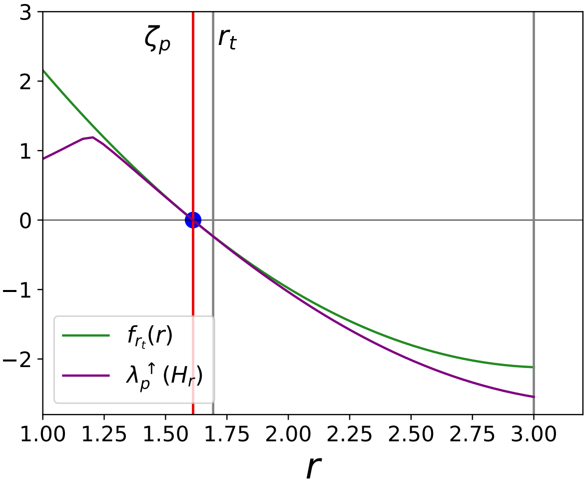
Efficient estimate of ζ_p

$\zeta_p \leftarrow r \ : \ \lambda_p^{\uparrow}(H_r) = 0$: grid search is **unefficient**



Efficient estimate of ζ_p

$\zeta_p \leftarrow r \; : \; \lambda_p^{\uparrow}(H_r) = 0$: grid search is **unefficient**



DCSBM graphs

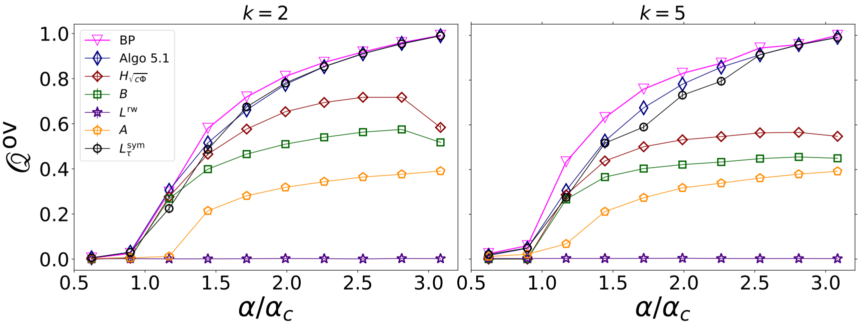


Figure: Performance vs competing methods on DCSBM synthetic graphs. $n = 50.000$, $c = 5$.

Real graphs

Dataset	n	c	k	H_{ζ_p}	A	$H_{\sqrt{c\Phi}}$	B	L^{rw}	L_{τ}^{sym}
Karate	34	4.6	2	**0.37**	**0.37**	**0.37**	**0.37**	0.36	**0.37**
Dolphins	62	5	2	**0.38**	0.21	0.34	0.22	**0.38**	**0.38**
Polbooks	105	8.4	3	**0.50**	0.47	**0.50**	0.45	**0.50**	**0.50**
Football	115	10.7	12	**0.60**	**0.60**	**0.60**	**0.60**	**0.60**	**0.60**
Mail	1133	9.6	21	**0.52**	0.32	0.40	0.37	0.48	**0.52**
Polblogs	1222	27,4	2	**0.43**	0.25	0.27	0.23	0.00	**0.43**
Tv	3892	8.9	41	**0.85**	0.60	0.56	0.55	0.55	0.80
Facebook	4039	43.7	55	**0.76**	0.42	0.49	0.48	0.70	0.58
GrQc	4158	6.5	29	**0.80**	0.52	0.51	0.51	0.34	**0.80**
Power grid	4941	2.7	25	**0.92**	0.18	0.33	0.31	**0.92**	0.85
Politicians	5908	14.1	62	**0.82**	0.48	0.54	0.51	0.74	0.74
GNutella P2P	6299	6.6	4	**0.40**	0.20	0.14	0.14	0.00	0.35
Wikipedia	7066	28.3	22	0.27	0.14	0.18	0.16	**0.34**	0.27
HepPh	11204	21.0	60	**0.57**	0.46	0.42	0.42	0.27	0.52
Vip	11565	11.6	53	**0.62**	0.28	0.32	0.32	0.16	0.54

Table: Modularity (no ground truth) comparison on real networks

A unified framework

- Ising Hamiltonian $\rightarrow$ Bethe approximation $\rightarrow$ Hessian $\rightarrow$ H_r

A unified framework

- Ising Hamiltonian $\rightarrow$ Bethe approximation $\rightarrow$ Hessian $\rightarrow$ H_r
- Ising Hamiltonian $\rightarrow$ naïve mean field approximation $\rightarrow$ Hessian $\rightarrow$ A

A unified framework

- Ising Hamiltonian $\rightarrow$ Bethe approximation $\rightarrow$ Hessian $\rightarrow$ H_r
- Ising Hamiltonian $\rightarrow$ naïve mean field approximation $\rightarrow$ Hessian $\rightarrow$ A

Naïve mean field vs Bethe approximation

A unified framework

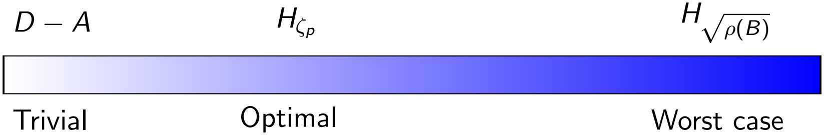
- Ising Hamiltonian $\rightarrow$ Bethe approximation $\rightarrow$ Hessian $\rightarrow$ H_r
- Ising Hamiltonian $\rightarrow$ naïve mean field approximation $\rightarrow$ Hessian $\rightarrow$ A

Naïve mean field vs Bethe approximation

- Unsuited for sparse graphs
- Does not reach α_c threshold on DCSBM graphs
- Does not keep heterogeneity into account

A unified framework

$$H_r = (r^2 - 1)I_n + D - rA$$



$r = \zeta_p$ is the hardness-dependent optimal regularization

A unified framework

$$L_\tau = D_\tau^{-1/2} A D_\tau^{-1/2}, \quad D_\tau = D + \tau I_n$$

A unified framework

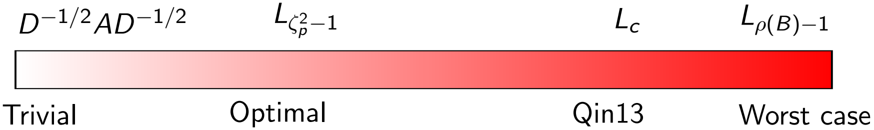
$$L_\tau = D_\tau^{-1/2} A D_\tau^{-1/2}, \quad D_\tau = D + \tau I_n$$

$$(D_{\zeta_p^2-1} - \zeta_p A)\boldsymbol{x}_p = 0 \Longrightarrow D_{\zeta_p^2-1}^{-1} A \boldsymbol{x}_p = \frac{1}{\zeta_p}\boldsymbol{x}_p$$

A unified framework

$$L_\tau = D_\tau^{-1/2} A D_\tau^{-1/2}, \quad D_\tau = D + \tau I_n$$

$$(D_{\zeta_p^2-1} - \zeta_p A)\boldsymbol{x}_p = 0 \Longrightarrow D_{\zeta_p^2-1}^{-1} A \boldsymbol{x}_p = \frac{1}{\zeta_p}\boldsymbol{x}_p$$



$\tau = \zeta_p^2 - 1$ is the hardness-dependent optimal regularization

Contents

CONCLUSION

Outlook

Two bridges

- Between "classical" and "physics-based" methods

Outlook

Two bridges

- Between "classical" and "physics-based" methods
- Between "sparse" and "dense" regimes

Outlook

Two bridges

- Between “classical” and “physics-based” methods
- Between “sparse” and “dense” regimes

A simple paradigm

- The matrix **adapts** to the hardness

Outlook

Two bridges

- Between “classical” and “physics-based” methods
- Between “sparse” and “dense” regimes

A simple paradigm

- The matrix **adapts** to the hardness
- The **optimal** regularization is **computed**

Outlook

Two bridges

- Between “classical” and “physics-based” methods
- Between “sparse” and “dense” regimes

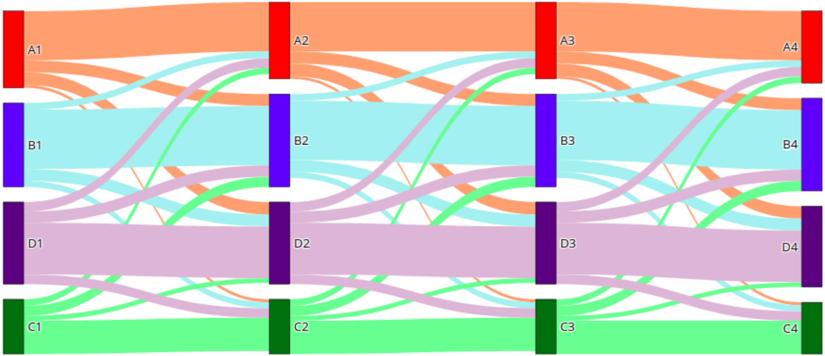
A simple paradigm

- The matrix **adapts** to the hardness
- The **optimal** regularization is **computed**

A question

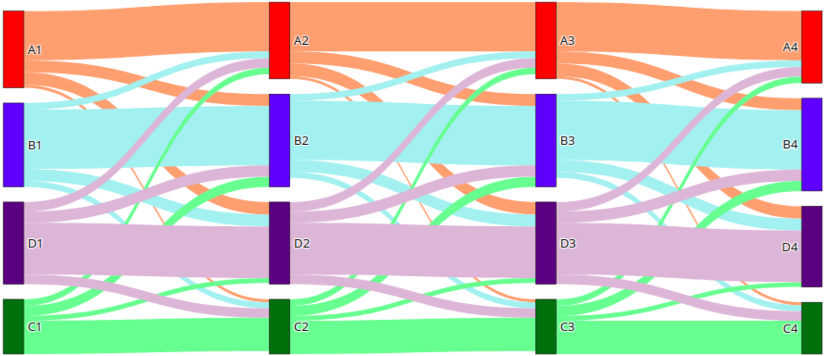
How to find the “right temperature” for a given ML problem?

Extensions: CD in dynamical graphs



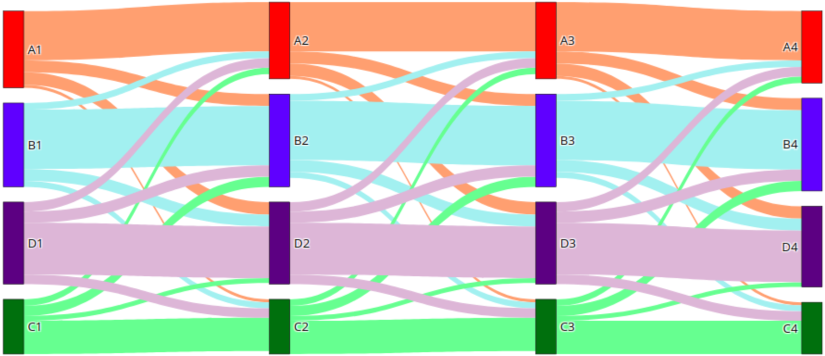
- Consider a sequence of graphs $\{\mathcal{G}_t\}_{t=1,\ldots,T}$

Extensions: CD in dynamical graphs



- Consider a sequence of graphs $\{\mathcal{G}_t\}_{t=1,\dots,T}$
- Suppose the labels are correlated across time $\mathbb{P}(\ell_i^{(t)} = \ell_i^{(t+1)}) = \eta$

Extensions: CD in dynamical graphs



- Consider a sequence of graphs $\{\mathcal{G}_t\}_{t=1,\dots,T}$
- Suppose the labels are correlated across time $\mathbb{P}(\ell_i^{(t)} = \ell_i^{(t+1)}) = \eta$
- Infer the labels as a function of t, exploiting their time correlation

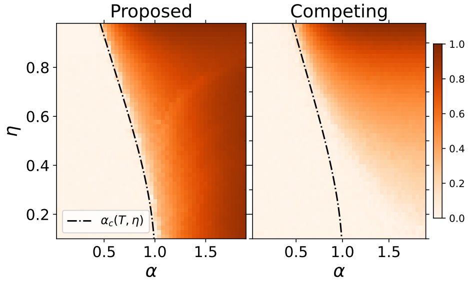
A "dynamical" Hamiltonian

$$\mathcal{H}(\boldsymbol{s}) = -\sum_{t=1}^{T}\sum_{(i_t,j_t)\in\mathcal{V}_t} \underbrace{\mathrm{ath}(\xi) s_{i_t} s_{j_t}}_{\text{Same } t} - \sum_{t=1}^{T-1}\sum_{i_t\in\mathcal{V}_t} \underbrace{\mathrm{ath}(h) s_{i_t} s_{i_{t+1}}}_{t \iff t+1}$$

A “dynamical” Hamiltonian

$$\mathcal{H}(\boldsymbol{s}) = -\sum_{t=1}^{T}\sum_{(i_t,j_t)\in\mathcal{V}_t} \underbrace{\mathrm{ath}(\xi)s_{i_t}s_{j_t}}_{\text{Same } t} - \sum_{t=1}^{T-1}\sum_{i_t\in\mathcal{V}_t} \underbrace{\mathrm{ath}(h)s_{i_t}s_{i_{t+1}}}_{t \iff t+1}$$

… and a dynamical Bethe-Hessian for SC



Extensions: relating Nishimori to Bethe

Erdős-Rényi random graph $\mathcal{G}$, $\forall\ (ij) \in \mathcal{E}$, $\omega_{ij} \sim P_0(|\omega_{ij}|)e^{\beta_{\rm N}\omega_{ij}}$

Extensions: relating Nishimori to Bethe

Erdős-Rényi random graph $\mathcal{G}$, $\forall\ (ij) \in \mathcal{E}$, $\omega_{ij} \sim P_0(|\omega_{ij}|)e^{\beta_{\rm N}\omega_{ij}}$

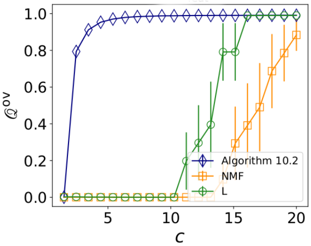
$$\hat{\beta}_{\rm N} = \max_{\beta}\{\beta\ :\ \lambda_1^{\uparrow}(H_\beta) = 0\}$$

$$\hat{\beta}_{\rm N} = \beta_{\rm N} + o_n(1)$$

Extensions: relating Nishimori to Bethe

Erdős-Rényi random graph $\mathcal{G}$, $\forall\ (ij) \in \mathcal{E}$, $\omega_{ij} \sim P_0(|\omega_{ij}|)e^{\beta_{\rm N}\omega_{ij}}$

$$\hat{\beta}_{\rm N} = \max_{\beta}\{\beta\ :\ \lambda_1^{\uparrow}(H_\beta) = 0\}$$
$$\hat{\beta}_{\rm N} = \beta_{\rm N} + o_n(1)$$



Cost-efficient **data** clustering

Publications

Journal papers

- **LD**, Romain Couillet Nicolas Tremblay: *Nishimori meets Bethe: a spectral method for node classification in sparse weighted graphs*, in Journal of statistichal mechanics
- **LD**, Romain Couillet, Nicolas Tremblay: *A unified framework for spectral clustering in sparse graphs*, in JMLR

Conference papers

- **LD**, Romain Couillet Nicolas Tremblay: *Community detection in sparse time-evolving graphs with a dynamical Bethe-Hessian*, in NeurIPS 2020
- **LD**, Romain Couillet, Nicolas Tremblay: *Optimal Laplacian regularization for sparse spectral community detection*, in ICASSP 2020
- **LD**, Romain Couillet, Nicolas Tremblay: *Classification spectrale par la laplacienne déformée dans des graphes réalistes*, in GRETSI 2019
- **LD**, Romain Couillet, Nicolas Tremblay: *Revisiting the Bethe-Hessian: improved community detection in sparse heterogeneous graphs*, in NeurIPS 2019
- **LD**, Romain Couillet: *Community Detection in Sparse Realistic Graphs: Improving the Bethe Hessian*, in ICASSP 2019