

# Community Detection in the Setting of Generalized Random Dot Product Graphs

John Koo

Department of Statistics  
Indiana University

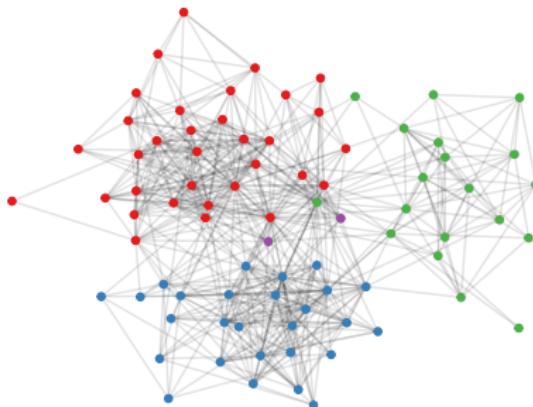
January 2023

# Outline

1. Random Graph Models
  - Block Models
  - Generalized Random Dot Product Graphs
2. PABMs are GRDPGs
3. Manifold Block Models

# Random Graph Models

## Problem Setup



Friendship network of 81 faculty at a UK university (Nepusz et al., 2008). The vertices are labeled by school affiliation.

How might we cluster the nodes of a network?

## Bernoulli Random Graphs

Let  $G$  be an undirected and unweighted graph with  $n$  vertices.

$G$  is described by adjacency matrix  $A$  such that

$$A_{ij} = \begin{cases} 1 & \text{an edge connects vertices } i \text{ and } j \\ 0 & \text{otherwise} \end{cases}$$

$$A_{ji} = A_{ij} \text{ and } A_{ii} = 0.$$

$A \sim \text{BernoulliGraph}(P)$  iff:

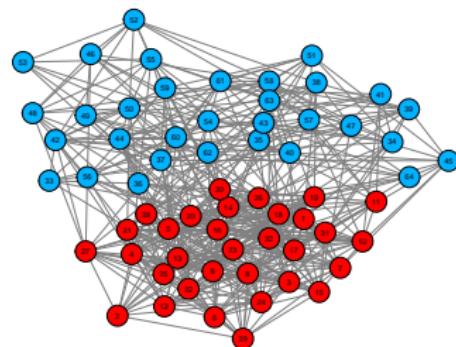
1.  $P$  is a matrix of edge probabilities between pairs of vertices.
2.  $A_{ij} \stackrel{\text{ind}}{\sim} \text{Bernoulli}(P_{ij})$  for each  $i < j$ .

## Block Models

Suppose each vertex  $v_1, \dots, v_n$  has labels  $z_1, \dots, z_n \in \{1, \dots, K\}$ , and each  $P_{ij}$  depends on labels  $z_i$  and  $z_j$ . Then  $A \sim \text{BernoulliGraph}(P)$  is a *block model*.

**Example 1:** Stochastic Block Model with  $K = 2$  communities.

$$P_{ij} = \begin{cases} p & z_i = z_j = 1 \\ q & z_i = z_j = 2 \\ r & z_i \neq z_j \end{cases}$$



# Block Models

## Erdos-Renyi Model (1959)

- $P_{ij} = \theta$  (not a block model)
- 1 parameter  $\theta$

## Stochastic Block Model (Lorrain and White, 1971)

- $P_{ij} = \theta_{z_iz_j}$
- $K(K + 1)/2$  parameters  $\theta_{k\ell}$

## Degree Corrected Block Model (Karrer and Newman, 2011)

- $P_{ij} = \theta_{z_iz_j} \omega_i \omega_j$
- $K(K + 1)/2 + n$  parameters  $\theta_{k\ell}, \omega_i$

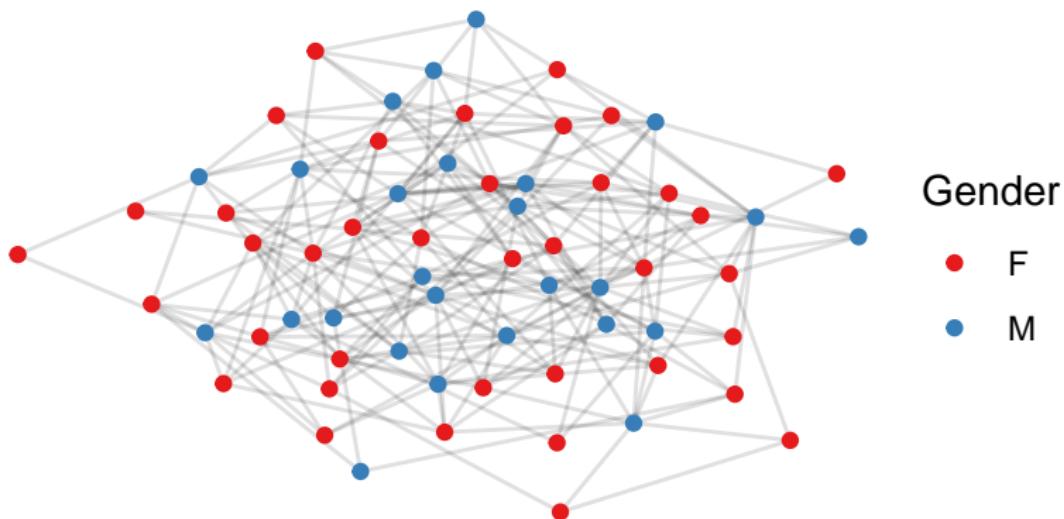
## Popularity Adjusted Block Model (Sengupta and Chen, 2017)

- $P_{ij} = \lambda_{iz_j} \lambda_{jz_i}$
- $Kn$  parameters  $\lambda_{ik}$

## Block Models

Dating network as an SBM:  $P_{ij} = \theta_{z_i, z_j}$

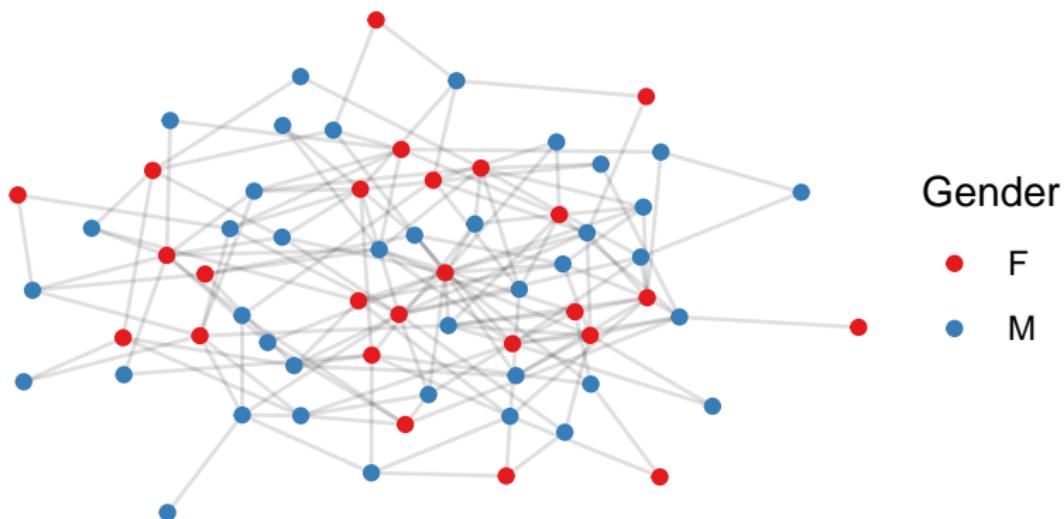
- This model is disassortative;  $\theta_{k\ell} > \theta_{kk}$  for  $k \neq \ell$ .
- Each individual has the same probability of matching with every other individual conditioned on gender.



## Block Models

Dating network as a DCBM:  $P_{ij} = \omega_i \omega_j \theta_{z_i, z_j}$

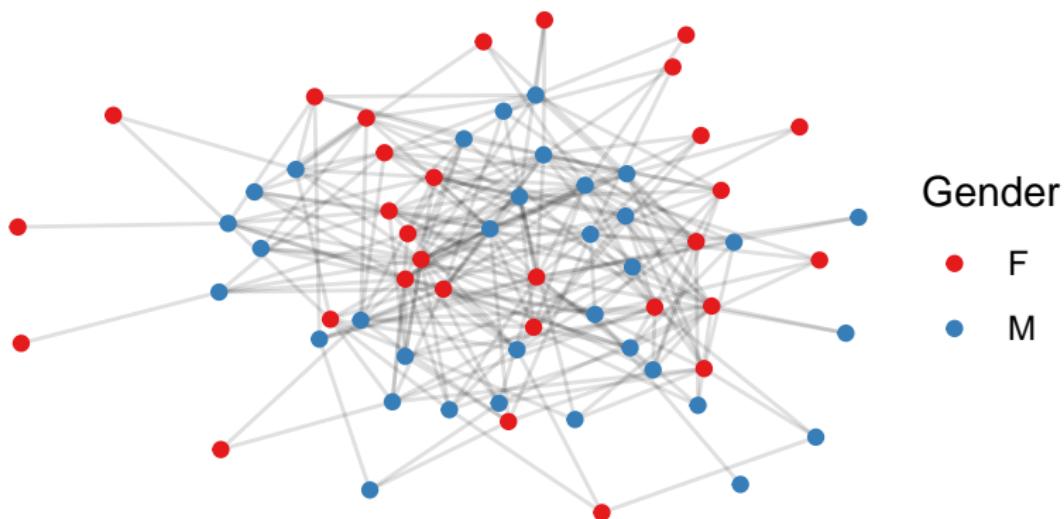
- Models each individual's popularity—some individuals form more connections than others.
- Still assumes that all members of each gender have the same odds of matching between two genders.



## Block Models

Dating network as a PABM:  $P_{ij} = \lambda_{i,z_j} \lambda_{j,z_i}$

- Models each user's sexual orientation as well as popularity.
- Some individuals are more likely to connect with members of the same gender while others are more likely to connect with members of the opposite gender.



# Hierarchy of Block Models

PABM  $\rightarrow$  DCBM:  $\lambda_{ik} = \sqrt{\theta_{z_i k}} \omega_i$

DCBM  $\rightarrow$  SBM:  $\omega_i = 1$

SBM  $\rightarrow$  Erdos-Renyi:  $\theta_{k\ell} = \theta$

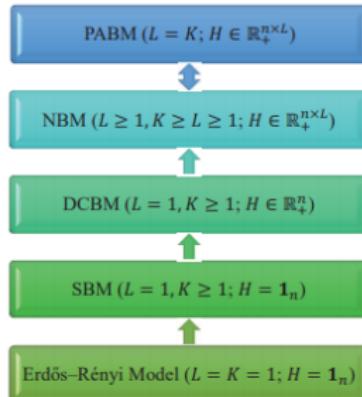


Figure 2: The hierarchy of block models

## Estimation for Block Models

Log-likelihood function for the SBM:

$$\ell(z, \theta; A) = \sum_{i < j} \sum_{k, \ell} z_{ik} z_{j\ell} \left( A_{ij} \log_{k\ell} + (1 - A_{ij}) \log(1 - \theta_{k\ell}) \right)$$

- Direct maximization is NP-hard.
- Expectation-Maximization requires an independence relaxation.

# Estimation for Block Models

---

## Algorithm 1: Approximate EM algorithm for the SBM

---

**Data:** Adjacency matrix  $A$ , number of communities  $K$

**Result:** Estimated community label probabilities  $\{\pi_{ik}\}$  for which each

$\pi_{ik} = P(z_i = k \mid A)$ , estimated community edge probabilities  $\{\hat{\theta}_{k\ell}\}_K$

```
1 Initialize  $\{\pi_{ik}\}$ ,  $\{\theta_{k\ell}\}$ .
2 while  $\|\nabla \ell\| > \epsilon$  do
3     for  $i = 1, \dots, n$  do
4         for  $k = 1, \dots, K$  do
5             E-step:
6                 
$$\pi_{ik} \propto \exp \left( \sum_{j \neq i} \sum_{\ell} \pi_{j\ell} (A_{ij} \log \hat{\theta}_{k\ell} + (1 - A_{ij}) \log (1 - \hat{\theta}_{k\ell})) \right).$$

7             end
8         end
9         for  $k = 1, \dots, K$  do
10            for  $\ell = 1, \dots, K$  do
11                M-step: 
$$\hat{\theta}_{k\ell} = \frac{\sum_{i < j} A_{ij} \pi_{ik} \pi_{j\ell}}{\sum_{i < j} \pi_{ik} \pi_{j\ell}}.$$

12            end
13        end
14    end
```

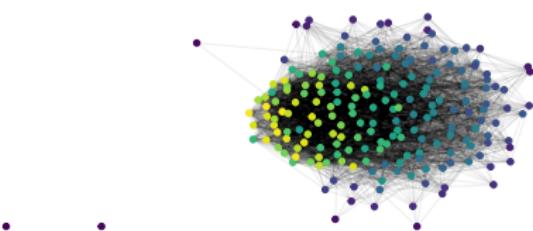
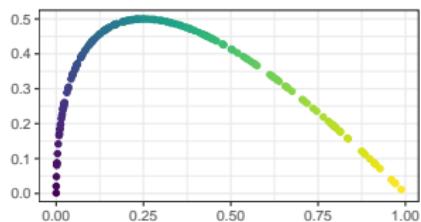
---

# Generalized Random Dot Product Graphs

**Def** Generalized Random Dot Product Graph  
(Rubin-Delanchy, Cape, Tang, Priebe, 2022)

Let  $I_{p,q} = \text{blockdiag}(I_p, -I_q)$  and suppose that  $x_1, \dots, x_n \in \mathbb{R}^{p+q}$  are such that  $x_i^\top I_{p,q} x_j \in [0, 1]$ .

Then  $A \sim \text{GRDPG}_{p,q}(X)$  iff  $A \sim \text{BernoulliGraph}(XI_{p,q}X^\top)$ ,  
where  $X = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^\top$ .



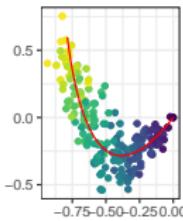
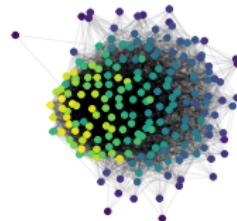
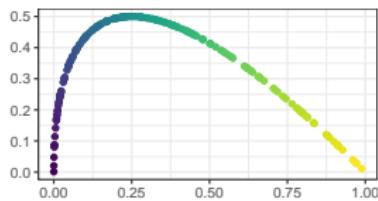
Latent vectors (left) and GRDPG (right).

# Generalized Random Dot Product Graphs

Adjacency Spectral Embedding (Sussman et al., 2012) estimates  $x_1, \dots, x_n \in \mathbb{R}^{p+q}$  from  $A$ :

1. Let  $\hat{\Lambda}$  be the diagonal matrix that contains the absolute values of the  $p$  most positive and the  $q$  most negative eigenvalues.
2. Let  $\hat{V}$  be the matrix whose columns are the corresponding eigenvectors.
3. Compute  $\hat{X} = \hat{V}\hat{\Lambda}^{1/2}$ .

**Theorem:**  $\max_i \|\hat{X}_i - Q_n X_i\| = O_P\left(\frac{(\log n)^c}{n^{1/2}}\right)$  as  $n \rightarrow \infty$



Latent vectors (left), GRDPG (center), and ASE (right).

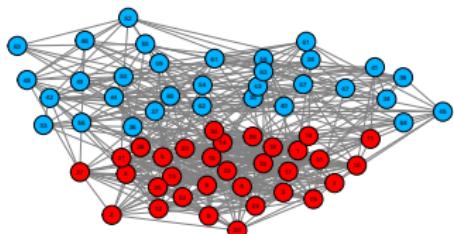
# SBMs are GRDPGs

All Bernoulli Graphs are GRDPGs.

**Example 1** (cont'd): SBM with  $K = 2$ .

$$P_{ij} = \begin{cases} p & z_i = z_j = 1 \\ q & z_i = z_j = 2 \\ r & z_i \neq z_j \end{cases}$$

$$P = \begin{bmatrix} P^{(11)} & P^{(12)} \\ P^{(21)} & P^{(22)} \end{bmatrix} = X I_{2,0} X^\top$$

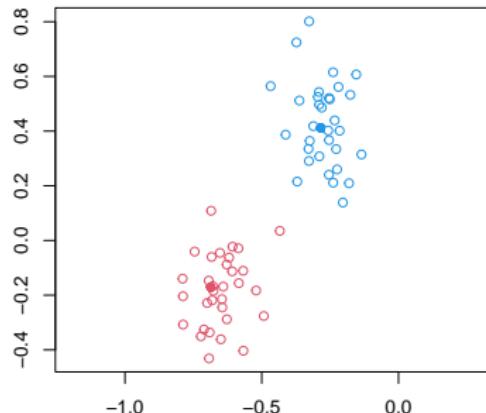


$$X = \begin{bmatrix} \sqrt{p} & 0 \\ \vdots & \vdots \\ \sqrt{p} & 0 \\ \sqrt{r^2/p} & \sqrt{q - r^2/p} \\ \vdots & \vdots \\ \sqrt{r^2/p} & \sqrt{q - r^2/p} \end{bmatrix}$$

## SBMs are GRDPGs

**Example 1** (cont'd): To perform community detection,

1. Note that  $A$  is a GRDPG because  $P = XI_{2,0}X^\top$ .
2. Compute the ASE  $A \approx \hat{X}\hat{X}^\top$  with  $\hat{X} = \hat{V}\hat{\Lambda}^{1/2}$ .
3. Apply a clustering algorithm (e.g.,  $K$ -means) to  $\hat{X}$ , noting that  $\hat{X}$  approaches point masses as  $n \rightarrow \infty$ .



# DCBMs are GRDPGs

## Example 2

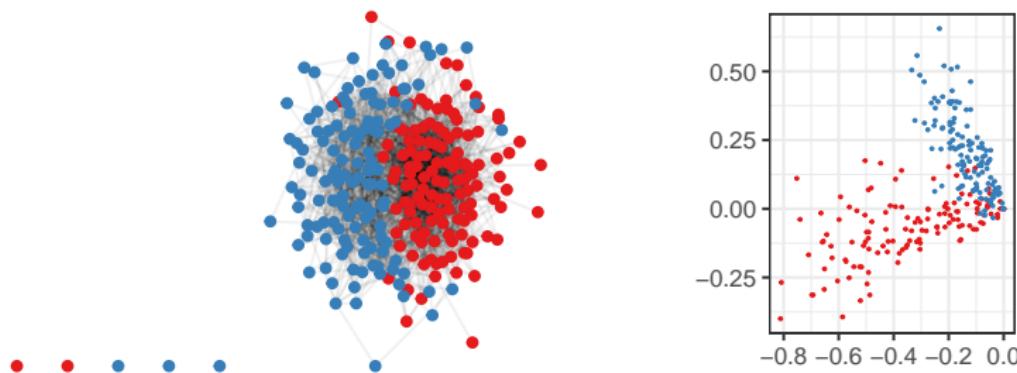
- Convert example 1 to a DCBM by  $P_{ij} \leftarrow \omega_i \omega_j P_{ij}$ .
- This is equivalent to  $P \leftarrow \Omega P \Omega$  where  $\Omega = \text{diag}(\omega_1, \dots, \omega_n)$ .
- This is equivalent to left-multiplying the matrix of latent vectors  $X$  by  $\Omega$ .
- The point masses in the latent structure of the SBM are “stretched out” toward the origin to form rays.

$$X = \begin{bmatrix} \omega_1 \sqrt{p} & 0 \\ \vdots & \vdots \\ \omega_{n_1} \sqrt{p} & 0 \\ \omega_{n_1+1} \sqrt{r^2/p} & \omega_{n_1+1} \sqrt{q - r^2/p} \\ \vdots & \vdots \\ \omega_n \sqrt{r^2/p} & \omega_n \sqrt{q - r^2/p} \end{bmatrix}$$

## DCBMs are GRDPGs

**Example 2** (cont'd): To perform community detection,

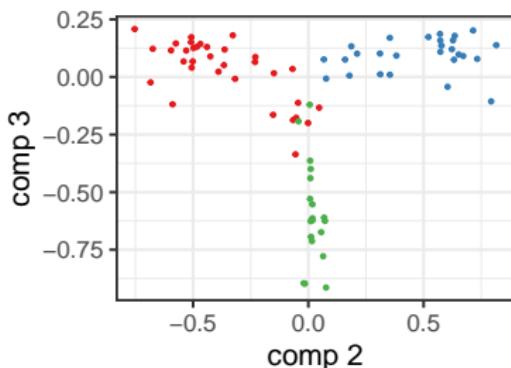
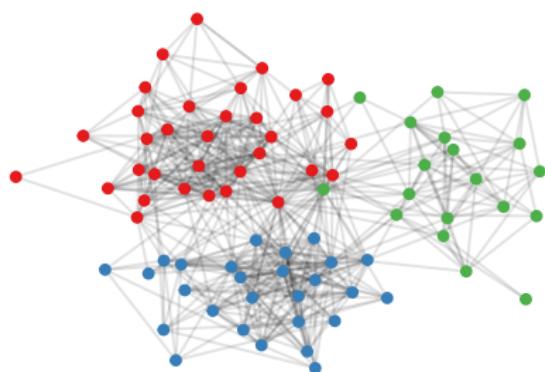
1. Note that  $A$  is a GRDPG because  $P = XI_{2,0}X^\top$ .
2. Compute the ASE  $A \approx \hat{X}\hat{X}^\top$  with  $\hat{X} = \hat{V}\hat{\Lambda}^{1/2}$ .
3. Apply a clustering algorithm (e.g., kernel  $K$ -means with the cosine kernel) to  $\hat{X}$ , noting that  $\hat{X}$  approaches rays as  $n \rightarrow \infty$ .



DCBM (left) and its ASE (right).

## Example: UK Faculty Network

- Treating graphs with community structure as a GRDPG is useful for both EDA and inference.
- The ASE of the UK faculty network (Nepusz et al., 2008) reveals three communities and that a DCBM is an appropriate model.
- Clustering on the ASE results in low community detection error.



UK faculty network (left) and its ASE (right).

PABMs are GRDPGs

## Popularity Vectors

**Lemma** (Noroozi, Rimal, and Pensky, 2020):

$A$  is sampled from a PABM if  $P$  can be described as:

1. Let each  $P^{(k\ell)}$  denote the  $n_k \times n_l$  matrix of edge probabilities between communities  $k$  and  $l$ .
2. Organize popularity parameters as vectors  $\lambda^{(k\ell)} \in \mathbb{R}^{n_k}$  such that  $\lambda_i^{(k\ell)} = \lambda_{k_il}$  is the popularity parameter of the  $i^{\text{th}}$  vertex of community  $k$  towards community  $l$ .
3. Each block can be decomposed as  $P^{(k\ell)} = \lambda^{(k\ell)}(\lambda^{(lk)})^\top$ .

This result reveals that  $P$  is rank  $K^2$  and has a linear latent structure.

## PABMs are GRDPGs

**Theorem** (KTT):  $A \sim \text{PABM}(\{\lambda_{ik}\}_K)$  is equivalent to  $A \sim \text{GRDPG}_{p,q}(XU)$  with

- $p = K(K + 1)/2$ ,  $q = K(K - 1)/2$ ;
- $U$  is an orthogonal matrix;
- $X \in \mathbb{R}^{n \times K^2}$  is a block diagonal matrix composed of popularity vectors with each block corresponding to a community.

$$X = \begin{bmatrix} \Lambda^{(1)} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \Lambda^{(K)} \end{bmatrix} \in \mathbb{R}^{n \times K^2}$$

$$\Lambda^{(k)} = \begin{bmatrix} \lambda^{(k1)} & \dots & \lambda^{(kK)} \end{bmatrix} \in \mathbb{R}^{n_k \times K}$$

$A \sim \text{PABM}(\{\lambda_{ik}\}_K)$  iff  $A \sim \text{GRDPG}_{p,q}(XU)$

## PABMs are GRDPGs

$$X = \begin{bmatrix} \Lambda^{(1)} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \Lambda^{(K)} \end{bmatrix} \quad U \in \mathbb{O}(K^2)$$

$$A \sim \text{PABM}(\{\lambda^{(k\ell)}\}_K) \iff A \sim \text{GRDPG}_{p,q}(XU)$$

**Remark 1** (orthogonality of subspaces): If  $y_i^\top$  and  $y_j^\top$  are two rows of  $XU$  corresponding to different communities, then  $y_i^\top y_j = 0$ .

**Remark 2** (non-uniqueness of the latent configuration):

If  $A \sim \text{GRDPG}_{p,q}(Y)$ , then  $A \sim \text{GRDPG}_{p,q}(YQ)$  for any  $Q$  in the indefinite orthogonal group with signature  $p, q$ .

**Remark 3:** Communities correspond to subspaces even with linear transformation  $Q \in \mathbb{O}(p, q)$ , but this may break the orthogonality property.

# Orthogonal Spectral Clustering

**Theorem** (KTT): If  $P = V\Lambda V^\top$  and  $B = nVV^\top$ , then  $B_{ij} = 0$  if  $z_i \neq z_j$ .

**Algorithm:** Orthogonal Spectral Clustering:

1. Let  $V$  be the eigenvectors of  $A$  corresponding to the  $K(K + 1)/2$  most positive and  $K(K - 1)/2$  most negative eigenvalues.
2. Compute  $B = |nVV^\top|$  applying  $|\cdot|$  entry-wise.
3. Construct graph  $G$  using  $B$  as its similarity matrix.
4. Partition  $G$  into  $K$  disconnected subgraphs.

# Orthogonal Spectral Clustering

**Theorem (KTT):**

Let  $\hat{B}$  with entries  $\hat{B}_{ij}$  be the affinity matrix from OSC. Then  $\forall$  pairs  $(i, j)$  belonging to different communities and sparsity factor satisfying  $n\rho_n = \omega((\log n)^{4c})$ ,

$$\max_{i,j} \hat{B}_{ij} = O_P\left(\frac{(\log n)^c}{\sqrt{n\rho_n}}\right) \text{ as } n \rightarrow \infty.$$

**Corollary:** OSC results in zero clustering error as  $n \rightarrow \infty$ , with probability 1.

# Sparse Subspace Clustering

**Corollary:** The ASE of  $A \sim \text{PABM}(\{\lambda^{(k\ell)}\}_K)$  lies near a collection of  $K$ -dimensional subspaces in  $K^2$  dimensions.

**Algorithm:** Sparse Subspace Clustering (Elhamifar & Vidal, 2009):

1. Solve  $n$  optimization problems  $c_i = \arg \min_c \|c\|_1$  subject to  $x_i = X^\top c$  and  $c^{(i)} = 0$ .

This is typically performed via LASSO:

$$c_i = \arg \min \frac{1}{2} \|x_i - X_{-i}^\top c\|_2^2 + \lambda \|c\|_1$$

2. Compile solutions  $C = [c_1 \quad \cdots \quad c_n]$ .
3. Construct affinity matrix  $B = |C| + |C^\top|$ .

## Sparse Subspace Clustering

Noroozi et al. observed that the rank of  $P$  is  $K^2$  and the columns of  $P$  belonging to each community have rank  $K$  to justify SSC for the PABM.

$$c_i = \arg \min_c \|c\|_1 \text{ subject to } A_{\cdot, i} = Ac \text{ and } c^{(i)} = 0$$

They were able to show that this obeys SDP if we replace  $A$  with  $P$ .

GRDPG-based approach: Apply SSC to the ASE of  $A$ .

Stronger result: Apply SSC to the eigenvectors of  $A$ .

$$c_i = \arg \min_c \|c\|_1 \text{ subject to } \hat{v}_i = \hat{V}^\top c \text{ and } c^{(i)} = 0$$

$$A \approx \hat{V} \hat{\Lambda} \hat{V}^\top$$

# Sparse Subspace Clustering

**Theorem (KTT):**

Let

- $P_n$  describe the edge probability matrix of the PABM with  $n$  vertices, and  $A_n \sim \text{BernoulliGraph}(P_n)$ ;
- $\hat{V}_n$  be the matrix of eigenvectors of  $A_n$  corresponding to the  $K(K + 1)/2$  most positive and  $K(K - 1)/2$  most negative eigenvalues.

Then

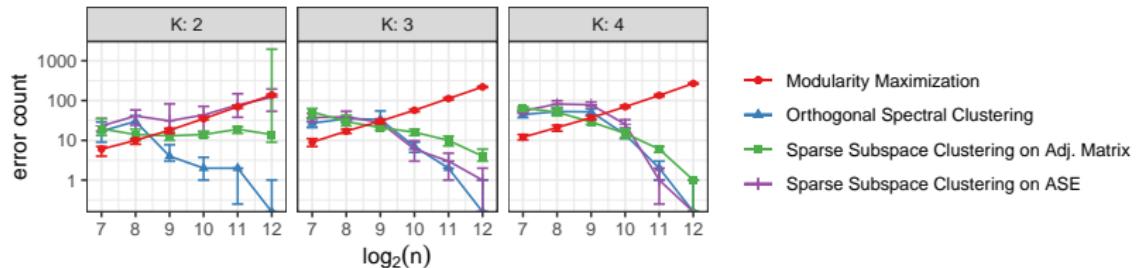
- For some  $\lambda > 0$  and  $N < \infty$ ,  $\sqrt{n}\hat{V}_n$  obeys the Subspace Detection Property with probability 1 when  $n > N$ .

Remarks:

- For large  $n$ , we can identify  $\lambda$  for SDP (Wang and Xu, 2016).
- SDP does not guarantee community detection.

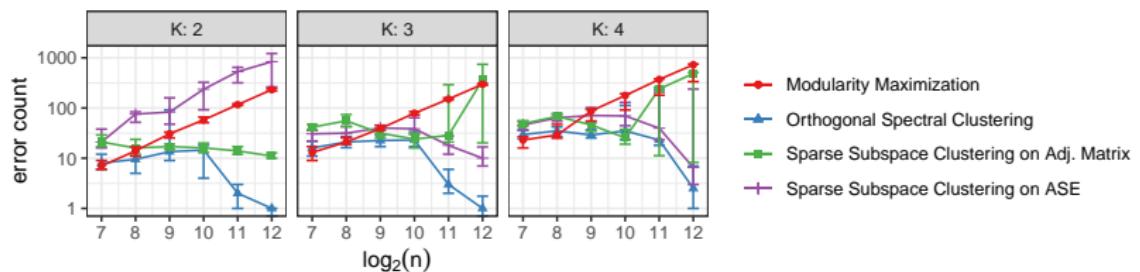
# Simulation Results

1.  $z_1, \dots, z_n \stackrel{\text{iid}}{\sim} \text{Categorical}(1/K, \dots, 1/K)$
2.  $\lambda_{ik} \stackrel{\text{iid}}{\sim} \text{Beta}(a_{ik}, b_{ik})$   
$$a_{ik} = \begin{cases} 2 & z_i = k \\ 1 & z_i \neq k \end{cases}, b_{ik} = \begin{cases} 1 & z_i = k \\ 2 & z_i \neq k \end{cases}$$
3.  $P_{ij} = \lambda_{iz_j} \lambda_{jz_i}$
4.  $A \sim \text{BernoulliGraph}(P)$



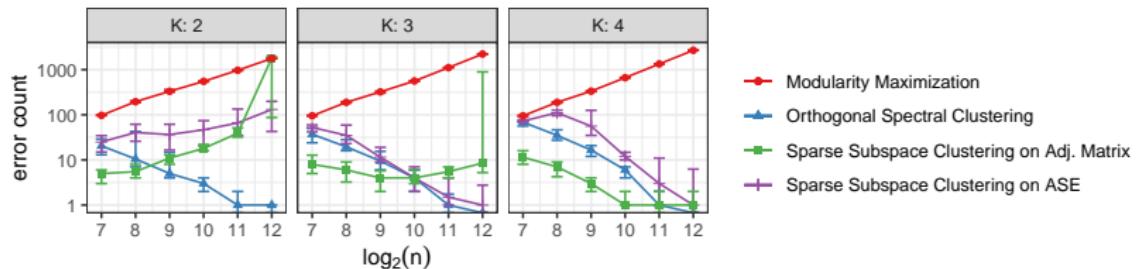
# Simulation Results

1.  $z_1, \dots, z_n \stackrel{iid}{\sim} \text{Categorical}(\alpha_1, \dots, \alpha_K)$  for  $\alpha_k = \frac{k^{-1}}{\sum_{\ell=1}^K \ell^{-1}}$
2.  $\lambda_{ik} \stackrel{iid}{\sim} \text{Beta}(a_{ik}, b_{ik})$   
$$a_{ik} = \begin{cases} 2 & z_i = k \\ 1 & z_i \neq k \end{cases}, b_{ik} = \begin{cases} 1 & z_i = k \\ 2 & z_i \neq k \end{cases}$$
3.  $P_{ij} = \lambda_{iz_j} \lambda_{jz_i}$
4.  $A \sim \text{BernoulliGraph}(P)$

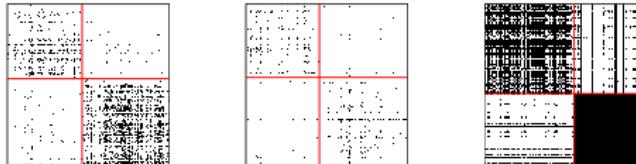


# Simulation Results

1.  $z_1, \dots, z_n \stackrel{\text{iid}}{\sim} \text{Categorical}(1/K, \dots, 1/K)$
2.  $\lambda_{ik} \stackrel{\text{iid}}{\sim} \text{Beta}(a_{ik}, b_{ik})$   
$$a_{ik} = \begin{cases} 1 & z_i = k \\ 2 & z_i \neq k \end{cases}, b_{ik} = \begin{cases} 2 & z_i = k \\ 1 & z_i \neq k \end{cases}$$
3.  $P_{ij} = \lambda_{iz_j} \lambda_{jz_i}$
4.  $A \sim \text{BernoulliGraph}(P)$



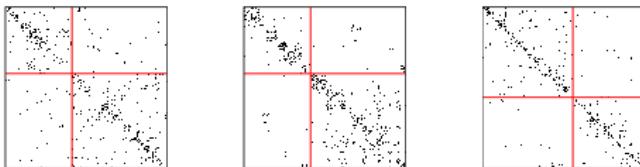
## Example



Adjacency matrices of (from left to right) the British MPs, Political Blogs, and DBLP networks after sorting by the clustering outputted by OSC.

Network	MM	SSC-ASE	OSC
British MPs	0.003	0.012	0.006
Political Blogs	0.050	0.187	0.062
DBLP	0.028	0.072	0.059

## Example: Karantaka Villages

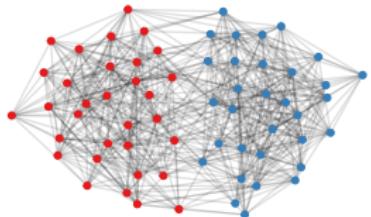


Adjacency matrix of the Karnataka villages data, arranged by the clustering produced by OSC (left). The villages studied here are, from left to right, 12, 31, and 46.

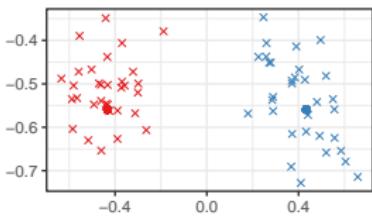
Network	MM	SSC-ASE	OSC
Village 12	0.270	0.291	0.227
Village 31	0.125	0.059	0.051
Village 46	0.052	0.069	0.056

# Overview of Block Models as GRDPGs

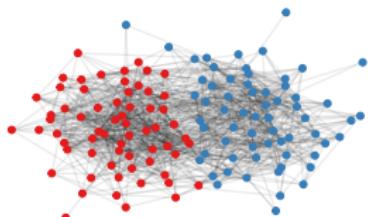
SBM



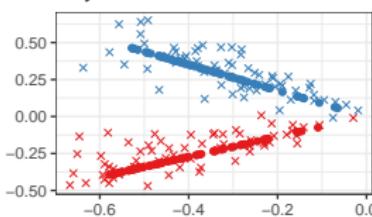
Point Masses



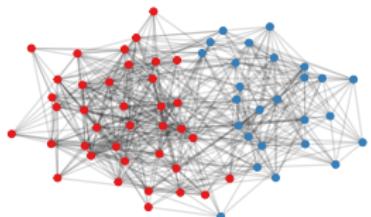
DCBM



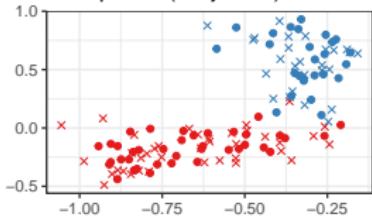
Rays



PABM



Subspaces (Projected)



- K-means clustering
- Gaussian mixture models

- K-means with cosine similarity
- GMM on angles

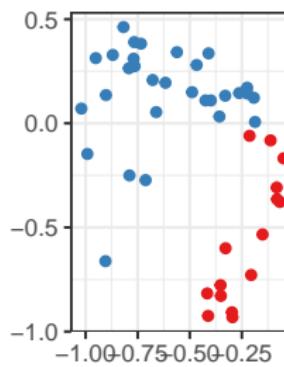
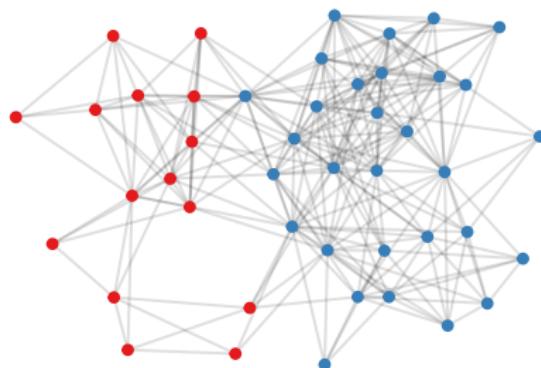
- Orthogonal spectral clustering
- Sparse subspace clustering

## Manifold Block Models

## Motivating Example

Macaque visuotactile brain areas and connections network  
(Negyessy et al., 2006)

- ASE suggests community structure.
- Structure is nonlinear— $K$ -means, OSC, SSC, etc. are not appropriate for these data.



Macaque visuotactile brain areas and connections network (left) and ASE (right). The vertices represent brain areas.

## Manifold Block Model

Let  $p, q \geq 0$ ,  $d = p + q \geq 1$ ,  $1 \leq r < d$ ,  $K \geq 2$ , and  $n > K$  be integers. Define manifolds  $\mathcal{M}_1, \dots, \mathcal{M}_K \in \mathcal{X}$  for

$\mathcal{X} = \{x, y \in \mathbb{R}^d : x^\top I_{p,q} y \in [0, 1]\}$  each by continuous function  $g_k : [0, 1]^r \rightarrow \mathcal{X}$ . Define probability distributions  $F_1, \dots, F_K$  each with support  $[0, 1]^r$ . Then the following mixture model is a *manifold block model*:

1. Draw labels  $z_1, \dots, z_n \stackrel{\text{iid}}{\sim} \text{Multinomial}(\alpha_1, \dots, \alpha_K)$ .
2. Draw latent vectors by first taking each  $t_i \stackrel{\text{ind}}{\sim} F_{z_i}$  and then computing each  $x_i = g_{z_i}(t_i)$ .
3. Compile the latent vectors into data matrix  $X = [x_1 \mid \dots \mid x_n]^\top$  and define the adjacency matrix as  $A \sim \text{GRDPG}_{p,q}(X)$ .

# *K*-Curves Clustering

---

## Algorithm 2: *K*-curves clustering.

---

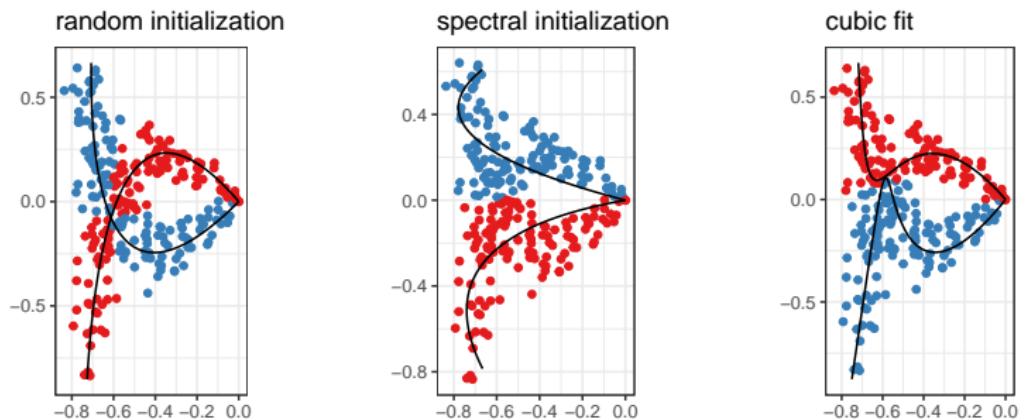
**Data:** Adjacency matrix  $A$ , number of communities  $K$ , embedding dimensions  $p$ ,  $q$ , stopping criterion  $\epsilon$

**Result:** Community assignments  $1, \dots, K$ , curves  $g_1, \dots, g_K$

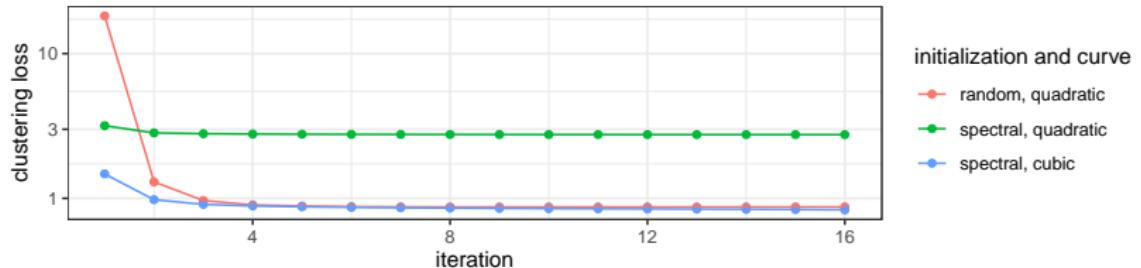
```
1 Compute  $X$ , the ASE of  $A$  using the  $p$  most positive and  $q$  most negative
   eigenvalues and their corresponding eigenvectors.
2 Initialize community labels  $z_1, \dots, z_n$ .
3 repeat
4   for  $k = 1, \dots, K$  do
5     Define  $X_k$  as the rows of  $X$  for which  $z_i = k$ .
6     Fit curve  $g_k$  and positions  $t_{k_i}$  to  $X_k$  by minimizing
          $\sum_{k_i} \|x_{k_i} - g_k(t_{k_i})\|^2$ .
7   end
8   for  $k = 1, \dots, K$  do
9     Assign  $z_i \leftarrow \arg \min_\ell \|x_i - g_\ell(t_i)\|^2$ .
10  end
11 until the change in  $\sum_k \sum_{i \in C_k} \|x_i - g_k(t_i)\|^2$  is less than  $\epsilon$ 
```

---

# *K*-Curves Clustering



*K*-curves clustering fits on the ASE using various approaches. Left: quadratic fit with random initialization. Middle: quadratic fit initialized via spectral clustering. Right: cubic fit initialized via spectral clustering.



## *K*-Curves Clustering

$$L(z_1, \dots, z_n, g_1, \dots, g_K; X) = \frac{1}{n} \sum_{k=1}^K \sum_{i:z_i=k} \|x_i - g_k(t_i)\|^2$$

**Theorem (KKT):**

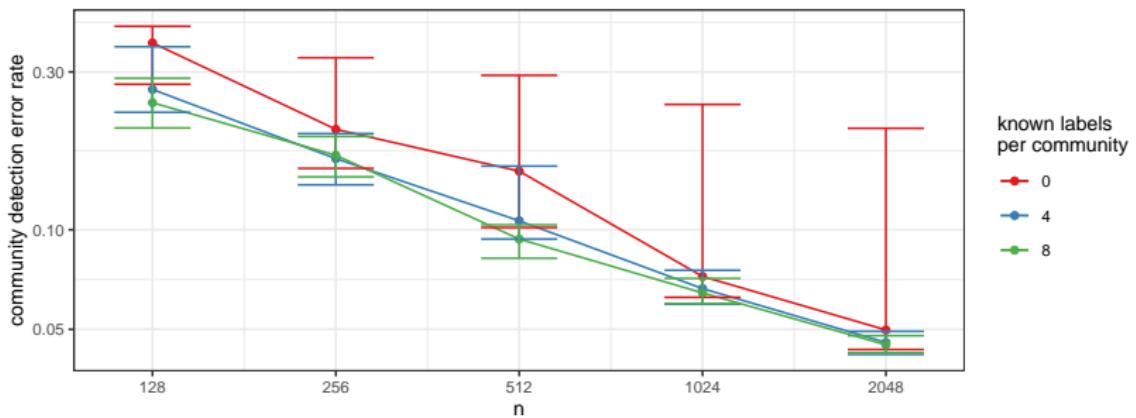
Let  $A \sim \text{MBM}(\{\alpha_1, \dots, \alpha_K\}, \{F, \dots, F\}, \{g_1, \dots, g_K\}; \rho_n)$  such that  $F$  has support  $[0, 1]$ , and each  $g_k(t) = g(t; p_k)$  is a curve of order  $R$  that does not self-intersect (for any  $s \neq t$ ,  $g_k(s) \neq g_k(t)$ ).

Suppose that for each community  $k$ , we have labels for at least  $R + 1$  vertices. Then if  $n\rho_n = \omega(\log^{4c} n)$ , as  $n \rightarrow \infty$ , the estimates outputted by  $K$ -curves clustering are such that

$$L(\hat{z}_1, \dots, \hat{z}_n, \hat{g}_1, \dots, \hat{g}_K; X) \xrightarrow{p} 0.$$

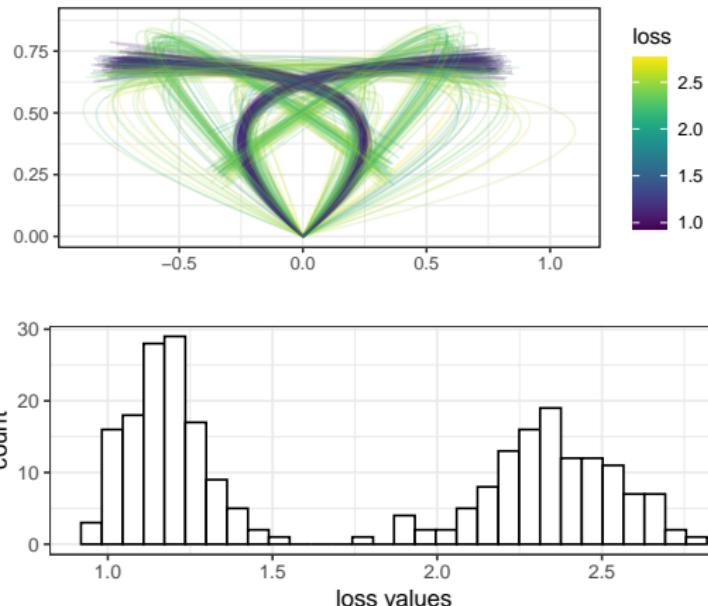
# Simulation Results

1. Draw  $z_1, \dots, z_n \stackrel{\text{iid}}{\sim} \text{Multinomial}(1/2, 1/2)$ .
2. Draw  $t_1, \dots, t_n \stackrel{\text{iid}}{\sim} \text{Uniform}(0, 1)$ .
3. Let each  $x_i = g_{z_i}(t_i)$  where  $g_1(t) = [t^2, 2t(1-t)]^\top$  and  $g_2(t) = [2t(1-t), (1-t)^2]^\top$ . Collect the latent vectors into matrix  $X = [x_1 | \dots | x_n]^\top$ .
4. Draw  $A \sim \text{RDPG}(X)$ .



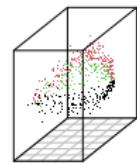
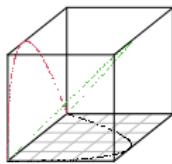
# Simulation Results

- Same setup as before but fix  $n = 512$  and only random initializations.
- Repeat simulation 256 times and look at the distribution of fitted curves and loss values.



## Simulation Results

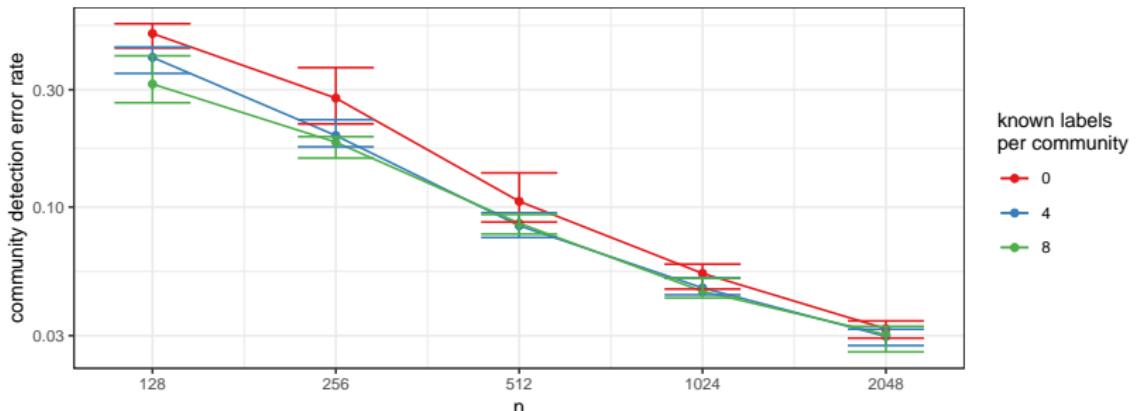
1. Draw  $z_1, \dots, z_n \stackrel{\text{iid}}{\sim} \text{Multinomial}(1/3, 1/3, 1/3)$ .
2. Draw  $t_1, \dots, t_n \stackrel{\text{iid}}{\sim} \text{Uniform}(0, 1)$ .
3. Let each  $x_i = g_{z_i}(t_i)$  where
  - i.  $g_1(t) = [2t(t-1), t^2, 0]^\top$
  - ii.  $g_2(t) = [0, t^2, 2t(t-1)]^\top$
  - iii.  $g_3(t) = [2t(t-1), t^2, 2t(t-1)]^\top$ .
4. Collect the latent vectors into matrix  $X = [x_1 \mid \dots \mid x_n]^\top$ .
5. Draw  $A \sim \text{RDPG}(X)$ .



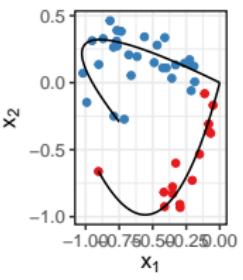
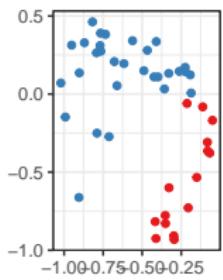
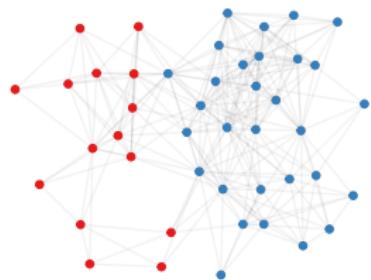
Latent vectors (left) and ASE (right) of one realization of the mixture model in the simulation setup with  $K = 3$  manifolds.

# Simulation Results

1. Draw  $z_1, \dots, z_n \stackrel{\text{iid}}{\sim} \text{Multinomial}(1/3, 1/3, 1/3)$ .
2. Draw  $t_1, \dots, t_n \stackrel{\text{iid}}{\sim} \text{Uniform}(0, 1)$ .
3. Let each  $x_i = g_{z_i}(t_i)$  where
  - i.  $g_1(t) = [2t(t-1), t^2, 0]^\top$
  - ii.  $g_2(t) = [0, t^2, 2t(t-1)]^\top$
  - iii.  $g_3(t) = [2t(t-1), t^2, 2t(t-1)]^\top$ .
4. Collect the latent vectors into matrix  $X = [x_1 \mid \dots \mid x_n]^\top$ .
5. Draw  $A \sim \text{RDPG}(X)$ .



## Example: Macaque Brain Areas Network



Macaque visuotactile brain areas and connections network (left), ASE (middle), and fitted curves (right). The vertices represent brain areas.

## Example: Drosophila connectome (Eichler et al., 2017)

