Community Detection in the Setting of Generalized Random Dot Product Graphs

John Koo

Submitted to the faculty of the Univesity Graduate School
in partial fulfillment of the requirements for the degree
Doctor of Philosophy
in the Department of Statistics,
Indiana University
December 2022

Accepted by the Graduate Faculty, Indiana University, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Approved:

Michael W. Trosset, Ph.D.

Minh Tang, Ph.D.

Julia Fukuyama, Ph.D.

Roni Khardon, Ph.D.

Fangzheng Xie, Ph.D.

December 1, 2022

Acknowledgements

Abstract

Graph and network data, in which samples are represented not as a collection of feature vectors but as relationships between pairs of observations, are increasingly widespread in various fields ranging from sociology to computer vision. One common goal of analyzing graph data is community detection or graph clustering, in which the graph is partitioned into disconnected subgraphs in an unsupervised yet meaningful manner (e.g., by optimizing an objective function or recovering unobserved labels). Because traditional clustering techniques were developed for data that can be represented as vectors, they cannot be applied directly to graphs. In this research, we investigate the use of a family of spectral decomposition based approaches for community detection in block models (random graph models with inherent community structure), first by demonstrating how under the Generalized Random Dot Product Graph framework, all graphs generated by block models can be represented as feature vectors, then applying clustering methods for these feature vector representations, and finally deriving the asymptotic properties of these methods.

Contents

1 Introduction

1.1 Graphs and Representations of Network Data

Graph and network data have become increasingly widespread in various fields including sociology, neuroscience, biostatistics, and computer science. This has resulted in challenges for researchers who rely on traditional statistical and machine learning methods that are incompatible with graph data and instead assume that the data exist as feature vectors. To illustrate this, consider the typical approach to building a statistical or machine learning model. Data are often represented as an $n \times p$ matrix $X = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^\top$ in which each row $x_i \in \mathbb{R}^p$ is an observation of p features and each column is a set of n feature measurements. An analysis task for these data might be to come up with a classification model $\phi : \mathbb{R}^p \to \{1, 2, ..., K\}$ that uses the numerical values of each feature of a vector x_i to calculate a predicted label $z_i \in \{1, 2, ..., K\}$. For instance, $\phi(x)$ might first compute the distances from x to K points in $\mathbb{R}^p$ and then assign x to the label of the nearest point. Examples of this include linear discriminant analysis (in the case of supervised learning) and Lloyd's algorithm (Lloyd, 1982) or Gaussian mixture models (Fraley and Raftery, 2002) (in the case of unsupervised learning). However, it is not obvious how this method would translate to data that are represented as graphs, in which observations consist of relationships among a set of objects rather than numerical attributes associated with each object: Instead of feature vector $x_i = \begin{bmatrix} x_{i1} & \cdots & x_{ip} \end{bmatrix}^\top \in \mathbb{R}^p$, we observe $a_i = \begin{bmatrix} a_{i1} & \cdots & a_{in} \end{bmatrix}^\top \in \mathbb{R}^n$ in which each a_{ij} is object i's relationship to object j. For such data, it is often necessary to alter existing algorithms, transform the graph data into Euclidean data, often called *graph embedding* or *spectral clustering* (von Luxburg, 2007), and apply algorithms for Euclidean data on the embedding, or come up with new algorithms altogether.

Example 1.1 (Graph clustering)**.** K-means clustering (MacQueen, 1967) is a nonparametric clustering method that minimizes the objective function

$$W(x_1, ..., x_n) = \sum_{k=1}^{K} \sum_{x_i : z_i = k} \|x_i - \bar{x}_k\|_2^2, \tag{1}$$

in which $x_1, ..., x_n \in \mathbb{R}^p$ are feature vectors of a sample, $z_1, ..., z_n \in \{1, ..., K\}$ are cluster assignments we wish to optimize for, and $\bar{x}_1, ..., \bar{x}_K \in \mathbb{R}^p$ are the centroids of each cluster, which act

as nuisance parameters. A popular implementation of K-means clustering is Lloyd's algorithm (Lloyd, 1982), which is a type of coordinate descent algorithm in which at each iteration labels are updated according to which centroid they are closest to and centroids are updated via the sample mean of each cluster.

If instead of vectors in $\mathbb{R}^p$ we have a graph, it is not obvious how to translate K-means clustering to these data. In particular, there is no inherent notion of centroid or sample mean for graph data. However, there are inherent notions of distances among vertices, such as shortest path distance, expected commute time, or resistance distance. One way to adapt K-means to graph data is to use an equivalent objective function as Eq. (1) that doesn't use centroids (which are just nuisance parameters) and is stated only in terms of distances:

$$\tilde{W}(x_1, ..., x_n) = \sum_{k=1}^{K} \sum_{x_i, x_j : z_i = z_j = k} \left(d(x_i, x_j)\right)^2.$$

Here, $d(x_i, x_j)$ is the distance between objects x_i and x_j. If they are vectors in Euclidean space, then $d(x_i, x_j) = \|x_i - x_j\|_2$, or if they are vertices on a graph, then $d(x_i, x_j)$ represents some notion of graph distance. While this formulation of K-means clustering takes care of the lack of centroids, the implementation still requires some thought since Lloyd's algorithm includes the computation of centroids. An alternative algorithm is MacQueen's exchange algorithm (MacQueen, 1967), which involves cycling through each vertex and then, for each vertex, cycling through the labels and choosing the label that minimizes the objective function.

Yet another approach to adapting K-means clustering to graph data is to embed the graph to Euclidean space and then apply Lloyd's algorithm on the embedding. One way to combine the graph distance approach and the graph embedding approach is to apply Lloyd's algorithm on an embedding that approximates graph distances (von Luxburg, 2007).

We now provide a more formal description of graph data: Suppose we observe a network of n objects and pairwise relationships between them. This network is represented by a graph $G = (V, E)$ with vertex set $V = \{v_1, ..., v_n\}$, representing the n objects, and edge set E, representing the up to $n(n-1)/2$ pairwise relationships (assuming that there are no self-loops). The numeric representation of these data is in the form of *affinity matrix* $A \in \mathbb{R}^{n \times n}$ in which each

A_{ij} represents object i's relationship to object j. We assume that the entries of A represent affinities or similarities, i.e., the higher the value of A_{ij}, the stronger the relationship i has to j. If $A_{ij} = 0$, then i has no direct relationship to j. A is symmetric if it represents an undirected graph in which the relationship from i to j is the same as the relationship from j to i. A is binary, i.e., $A \in \{0,1\}^{n \times n}$, if it represents an unweighted graph in which edges either exist or don't exist. If A is binary, we call it an *adjacency matrix*. In this work, we focus primarily on undirected and unweighted graphs without self loops.

Example 1.2. Nepusz et al. (2008) constructed a friendship network among 81 faculty from various schools at a university in the UK. These data are represented as a graph in which each vertex is a faculty member and edges between pairs of vertices indicate whether the two faculty are friends. The corresponding adjacency matrix $A \in \{0,1\}^{81 \times 81}$ has zeros along the diagonal since there are no self-loops, and $A_{ij} = A_{ji} = 1$ if it is observed that the i^{th} and j^{th} faculty members are friends. The following is a visualization of this graph, with the vertices labeled by school affiliation.

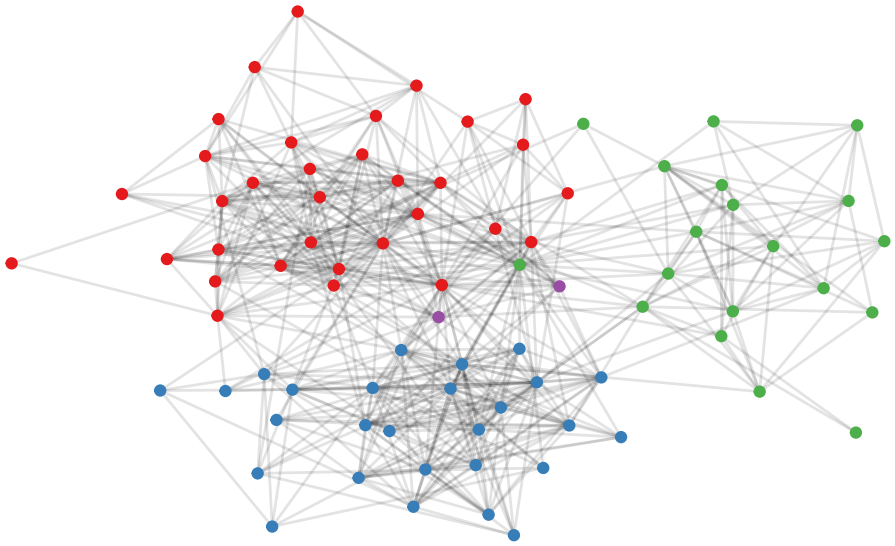

Figure 1: Friendship network of 81 faculty at a UK university. The vertices are labeled by school affiliation.

1.2 Probabilistic Models for Graphs

Given a sample or dataset, a typical analysis task is statistical inference, or the estimation of various parameters under the assumption that the data come from a random distribution or process.

These estimated parameters are often then used for making predictions or deriving insights about the population. For example, when fitting a Gaussian mixture model, the data are first assumed to come from a mixture of Gaussians. The model fitting process then involves estimating the means and standard deviations of each Gaussian component, along with the mixture weights. The resulting model provides insight into where each mixture component is located, how disperse each component is, and how the data are distributed between the components, as well as a prediction indicating to which mixture a new observation belongs. In order to perform a similar type of analysis for graphs, we must first define probability distributions from which such data can be sampled.

Within the scope of this work, we focus primarily on unweighted and undirected graphs without self-loops, with a brief discussion on generalizing these methods to weighted or directed graphs. The adjacency matrix that describes these graphs is binary, symmetric, and hollow. In this setting, a plausible model is to sample each edge independently from a Bernoulli distribution, i.e., $A_{ij} \overset{\text{ind}}{\sim} \text{Bernoulli}(P_{ij})$ for some $P_{ij} \in [0, 1]$ for each $i < j$ (setting $A_{ji} = A_{ij}$ since A is symmetric, and $A_{ii} = 0$ since A is hollow). Then similar to how the edges are compiled into an adjacency matrix A, the edge probabilities can be compiled into an edge probability matrix $P \in [0, 1]^{n \times n}$. This type of graph model is defined as a *Bernoulli graph* (also called an inhomogeneous Erdös-Rényi graph). If A is the adjacency matrix of a Bernoulli graph with edge probability matrix P, we denote $A \sim \text{BernoulliGraph}(P)$. If the vertices and edge probabilities are sampled as a sequence of n i.i.d. random variables from probability distribution F, then we denote $A \sim \text{BernoulliGraph}(F, n)$.

Statistical inference is not possible on a general Bernoulli graph with arbitrary edge probabilities since the number of parameters (individual edge probabilities) is equal to the number of observations (presence or absence of an edge). Additional structure must be introduced. One such structured Bernoulli graph model is the Erdös-Rényi graph, first proposed by Gilbert (1959), which is defined as follows:

Definition 1.1 (Erdös-Rényi graph)**.** Let P be an $n \times n$ matrix such that each $P_{ij} \equiv \theta \in [0, 1]$ is a constant. Then $A \sim \text{BernoulliGraph}(P)$ is an Erdös-Rényi graph.

Example 1.3 (Maximum likelihood estimator for the Erdös-Rényi graph)**.** One possible estimator

for the lone parameter of an Erdös-Rényi graph, θ, is the maximum likelihood estimator, which is found by maximizing the log-likelihood function $\ell(\theta; A) = \log\theta \sum_{i<j} A_{ij} + \log(1-\theta) \sum_{i<j}(1 - A_{ij})$. This can be solved directly by setting $\frac{d\ell}{d\theta} = 0$, which yields $\hat{\theta} = \frac{2|E|}{n(n-1)}$.

Erdös-Rényi graphs lie on the opposite end of the spectrum in that they restrict the model to a single parameter, which does not result in a very interesting model or reflect many networks observed in real data. Much work has been done in developing various Bernoulli graph models that strike a balance between structure and flexibility. Two common types of structure for graph models, that are of particular interest in this work, are edge probabilities based on latent communities (Lorrain and White, 1971, Airoldi et al., 2009, Karrer and Newman, 2011, Sengupta and Chen, 2018), called *block models*, and edge probabilities based on positions in a latent space (Young and Scheinerman, 2007, Rubin-Delanchy et al., 2017), called *latent space models*.

[Maybe insert something here about the exponential family of random graph models and how, like mixture distributions, block models do not fit within this family?]

Example 1.4. Consider a graph in which the vertices represent a collection of n planets with intelligent life sending out radio waves, and the existence of an edge between vertices i and j represents whether planets i and j have established contact. Since radio waves decay according to the inverse-square law, a plausible model for the edge probability between two vertices is

$$P_{ij} = \frac{C\omega_i\omega_j}{d_{ij}^2}$$

where ω_i is the i^{th} planet's radio signal strength, d_{ij} is the Euclidean distance between planets i and j, and C is a normalizing constant. Let P be the $n \times n$ matrix of these edge probabilities. Then $A \sim \text{BernoulliGraph}(P)$ represents the network of planets that have made contact.

To estimate $\omega = \begin{bmatrix}\omega_1 & \cdots & \omega_n\end{bmatrix}^\top$ after observing A and supposing that the relative locations of the planets as well as the normalizing constant C are known, one approach might be maximum likelihood maximization. For simplicity, set $C = 1$. Then the log-likelihood function can be written

as:

$$\ell(\omega; A) = \sum_{i<j} A_{ij} \log\left(\frac{\omega_i \omega_j}{d_{ij}^2 - \omega_i \omega_j}\right) + \log(d_{ij}^2 - \omega_i \omega_j) + const.$$

The partial derivatives with respect to each ω_i are:

$$\frac{\partial \ell}{\partial \omega_i} = \sum_{j \neq i} \frac{A_{ij}}{\omega_i} - \frac{\omega_j(1 - A_{ij})}{d_{ij}^2 - \omega_i \omega_j}.$$

Setting the gradient to zero yields n equations with n unknowns, but it is not entirely obvious how to solve this system of equations. Alternatively, we can use gradient ascent or some other numerical optimization method, since this particular log-likelihood function is concave.

Definition 1.2 (Degree of a vertex)**.** Let $G = (V, E)$ be an undirected graph with vertex set $V = \{v_1, ..., v_n\}$. d_i, the degree of vertex v_i is the sum of the weights of edges that connect to v_i. If G is unweighted, then v_i is equivalently the number of edges that connect to v_i. If G is represented by affinity or adjacency matrix A, then $d_i = \sum_j A_{ij}$.

If G is a Bernoulli graph with edge probability matrix P, then the expected degree of vertex v_i is $E[d_i] = \sum_j P_{ij}$.

1.3 Contributions of This Work

In this work, we explore three types of block models, the stochastic block model, the degree corrected block model, and the popularity adjusted block model, as well as a family of latent space models called generalized random dot product graphs. The contributions of this work are as follows. First, we show that, similar to how the stochastic block model and degree corrected block model are generalized random dot product graphs with specific latent structures, the popularity adjusted block model is also a generalized random dot product graph with a specific structure. Using estimators with well-established consistency properties for the generalized random dot product graph, we develop consistent community detection and parameter estimation algorithms for the popularity adjusted block model.

2 Block Models for Community Detection

Network analysis is often concerned with community detection. This has motivated statisticians to develop random graph models with inherent community structure in which the probability of an edge between vertices i and j depend on the communities to which vertices v_i and v_j belong. More formally, one assigns each vertex v_i a community label z_i and assumes a Bernoulli graph in which $P_{ij} = g(z_i, z_j, \theta_i, \theta_j)$ for some function $g : \{1, ..., K\}^2 \times \mathbb{R}^d \times \mathbb{R}^d \to [0, 1]$ and $\theta_i \in \mathbb{R}^d$ is a vector of real-valued parameters associated with each vertex. Within the context of this work, we will call such models *block models*. This type of model framework allows us to frame the goal of community detection as a statistical inference problem: identify the true community labels, up to permutation of the labels.

In this section, we explore the three main types of block models: the stochastic block model (Lorrain and White, 1971), the degree corrected block model (Karrer and Newman, 2011), and the popularity adjusted block model (Sengupta and Chen, 2018). The main focus is on how these models are connected, particularly in how they are nested models (Noroozi and Pensky, 2022), as well as community detection and parameter estimation via likelihood maximization. We will later return to these block models and the relationship to another class of random graph models in sections 3 and 4.

2.1 The Stochastic Block Model

The stochastic block model (SBM) (Lorrain and White, 1971) is the simplest of the three block models and assumes that each pair of communities (k, ℓ) has a constant edge probability $\theta_{k\ell}$. We now give the formal definition of the SBM within the context of Bernoulli graphs:

Definition 2.1 (Stochastic block model)**.** Let $G = (V, E)$ be a Bernoulli graph with n vertices, described by random adjacency matrix A. Let $K \geq 1$ be an integer and $\theta_{k\ell} \in [0, 1]$ for each $k, \ell \in \{1, ..., K\}$ (if G is undirected, then $\theta_{k\ell} = \theta_{\ell k}$). Let $z_1, ..., z_n \in \{1, ..., K\}$ be community labels associated with each vertex. If the edge probability matrix P for this graph is such that $P_{ij} = \theta_{z_i, z_j}$ and $A \sim \text{BernoulliGraph}(P)$, then G is a stochastic block model.

There are a few conventions for exactly how a graph is sampled as an SBM. Within the context

of this work, we assume that for a particular SBM, the number of communities, K, and the community edge probabilities, $\theta_{k\ell}$, are fixed. Then some possible ways of sampling from an SBM are as follows:

1. Let the size of the graph, n, be fixed. Then each vertex v_i, and its corresponding label, z_i, are also fixed, in addition to the pairwise community edge probabilities $\{\theta_{k\ell}\}_K$, which are all treated as model parameters. Under this sampling scheme, there is no notion of increasing n.
2. The vertices and edges are sampled in sequence, and we suppose that each vertex v_i comes with a corresponding label, z_i, which determines its edge probabilities to the other vertices. Under this sampling scheme, there is no assumed distribution on the labels, which are treated as fixed parameters rather than random variables, along with $\{\theta_{k\ell}\}_K$, but there is a notion of increasing sample size n, which allows for asymptotics.
3. The vertices are sampled in sequence, and each label, z_i, is sampled as $z_1, ..., z_n \overset{\text{iid}}{\sim} \text{Multinomial}(\alpha_1, ..., \alpha_K)$, $\sum_k^K \alpha_k = 1$. The edge probabilities are then determined by the sample $\{z_i\}$ and the parameters $\{\theta_{k\ell}\}_K$. Under this sampling scheme, both the edges and the labels are random, with fixed parameters $\alpha = \begin{bmatrix}\alpha_1 & \cdots & \alpha_K\end{bmatrix}^\top$ and $\{\theta_{k\ell}\}_K$.

We use the notation $A \sim \text{SBM}(z, \{\theta_{k\ell}\}_K)$ for the first two sampling schemes, and we use $A \sim \text{SBM}(\alpha, \{\theta_{k\ell}\}_K)$ for the third.

An extension to these include Bayesian models, e.g., each $\theta_{k\ell}$ is sampled from a distribution with support $[0, 1]$.

Example 2.1 (Stochastic block model with $K = 2$ communities)**.** We construct an SBM with two communities in which $\theta_{11} = 1/2$, $\theta_{12} = 1/8$, and $\theta_{22} = 1/4$. Note that in this example, the within-community edge probabilities are greater than the between-community edge probability. A realization of this graph with $n_1 = n_2 = 32$ (where $n_k = |\{v_i : z_i = k\}|$) is illustrated in figure 2.

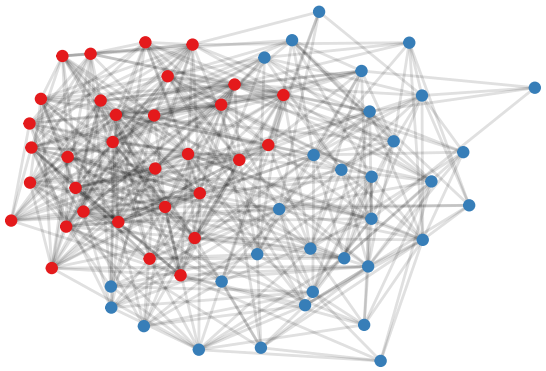

Figure 2: Two-community stochastic block model with $\theta_{11} = 1/2$, $\theta_{22} = 1/4$, and $\theta_{12} = \theta_{21} = 1/8$.

In most conceptions of block models, the within-community edge probability is greater than the between-community edge probability, as in example 2.1. For instance, in the UK faculty friendship network in example 1.2, it is natural to expect that faculty within the same school are more likely to be friends than faculty from two different schools, and a cursory look at the visualization of the network appears to confirm this. However, SBMs are not necessarily restricted to this type of community structure. We could just as easily construct an example in which θ_{11} and θ_{22} are less than θ_{12}. Within the context of block models, we say that a graph is *assortative* if P is positive semidefinite and *disassortative* otherwise, which roughly correspond to communities with higher within-community edge probabilities and lower between-community edge probabilities, respectively. For example, the SBM in example 2.1 is assortative and has edge probability matrix P that is rank 2 with nonzero eigenvalues $n(\frac{3}{8} \pm \frac{\sqrt{2}}{8}) > 0$. In the following example, we describe a network which can be modeled as a disassortative SBM.

Example 2.2 (Dating network as a disassortative stochastic block model)**.** Consider an undirected graph in which each vertex v_i represents users of an online dating service, each label z_i represents the user's gender, and each edge represents a successful match between pairs of users. For simplicity, we will restrict the labels to female and male (denoted as 1 and 2). If we model this as an SBM, then θ_{11} is the probability of a match between two female users, θ_{22} is the probability of a match between two male users, and $\theta_{12} = \theta_{21}$ is the probability of a match between a female user and a male user. Based on trends in the United States (Jones, 2022), a plausible set of edge probabilities is $\theta_{11} = \theta_{22} = 0.02$ and $\theta_{12} = 0.2$, which are used in the sample visualized in figure 3,

along with $\alpha_1 = \alpha_2 = 1/2$ and $n = 64$.

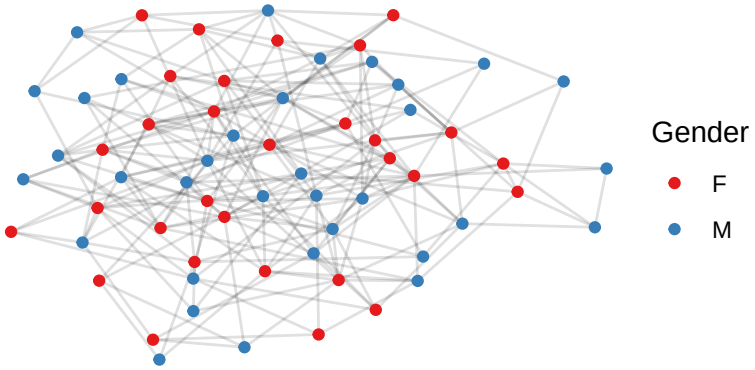

Figure 3: Stochastic block model of a dating network. This model is disassortative.

Example 2.3 (Assortativity and disassortativity of the two-community SBM)**.** Let P be the $n \times n$ probability matrix of a two-community SBM. Then $rank(P) = 2$, and its two nonzero eigenvalues are:

$$\lambda = \frac{n}{2}\left(\theta_{11} + \theta_{22} \pm \sqrt{(\theta_{11} - \theta_{22})^2 + 4\theta_{12}^2}\right).$$

The first eigenvalue is always positive. The second eigenvalue is positive if $\theta_{11}\theta_{22} > \theta_{12}^2$. Thus, P describes an assortative SBM if $\theta_{11}\theta_{22} > \theta_{12}^2$ or a disassortative SBM if $\theta_{11}\theta_{22} < \theta_{12}^2$.

One approach to statistical inference for the SBM is likelihood maximization. The log-likelihood function can be written as:

$$\ell(z, \theta; A) = \sum_{i<j} \sum_{k,\ell} z_{ik} z_{j\ell} \big(A_{ij} \log \theta_{k\ell} - (1 - A_{ij}) \log(1 - \theta_{k\ell})\big), \tag{2}$$

where $z_{ik} = I(z_i = k)$. If the labels are drawn from a multinomial distribution, then that can also be incorporated into the complete-data log-likelihood:

$$\ell(z, \theta, \alpha; A) = \sum_{k} \sum_{i} z_{ik} \log \alpha_k + \sum_{i<j} \sum_{k,\ell} z_{ik} z_{j\ell} \big(A_{ij} \log \theta_{k\ell} - (1 - A_{ij}) \log(1 - \theta_{k\ell})\big). \tag{3}$$

The SBM is a type of mixture model, and like most classical mixture models and clustering problems, likelihood maximization is NP-hard. Because the SBM is a mixture model, a candidate

algorithm for finding a (local) maximum of the likelihood is the expectation maximization (EM) algorithm (Dempster et al., 1977). However, due to the $z_{ik}z_{j\ell}$ interaction terms, the expectation step cannot be solved in closed form (Kolaczyk and Csárdi, 2014). While the labels are drawn independently, they are not necessarily conditionally independent. Nevertheless, making the conditional independence relaxation allows us to write the expectation and maximization steps in closed form. As an example, we write the EM algorithm for the likelihood function in equation (2) in algorithm 1.

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 $\{\hat{\theta}_{k\ell}\}_K$

Initialize $\{\pi_{ik}\}$, $\{\theta_{k\ell}\}$.
while $\|\nabla \ell\| > \epsilon$ **do**
  for $i = 1, ..., n$ **do**
    for $k = 1, ..., K$ **do**
      E-step: $\pi_{ik} \propto \exp\left(\sum_{j \neq i} \sum_{\ell} \pi_{j\ell} (A_{ij} \log \hat{\theta}_{k\ell} + (1 - A_{ij}) \log(1 - \hat{\theta}_{k\ell})) \right)$.
      M-step: $\hat{\theta}_{k\ell} = \frac{\sum_{i<j} A_{ij} \pi_{ik} \pi_{j\ell}}{\sum_{i<j} \pi_{ik} \pi_{j\ell}}$.
    end
  end
end

Applying algorithm 1 to example 2.1 with initial guesses of $\pi_{ik} = 0.5$ for each i, k, $\hat{\theta}_{11} = \hat{\theta}_{22} = 0.9$, and $\hat{\theta}_{12} = 0.1$ results in a community detection error rate of 4.7% and parameter estimates $\hat{\theta}_{11} = 0.503$, $\hat{\theta}_{22} = 0.241$, and $\hat{\theta}_{12} = 0.109$, compared to the true parameters $\theta_{11} = 0.5$, $\theta_{22} = 0.25$, and $\theta_{12} = 0.125$. The same algorithm applied to example 2.2, which is a disassortative model, using initial guesses of $\pi_{ik} = 0.5$, $\hat{\theta}_{11} = \hat{\theta}_{22} = 0.1$, and $\hat{\theta}_{12} = 0.9$, results in a community detection error rate of 0% and parameter estimates $\hat{\theta}_{11} = 0.034$, $\hat{\theta}_{22} = 0.018$, and $\hat{\theta}_{12} = 0.201$, compared to the true parameters $\theta_{11} = \theta_{22} = 0.02$ and $\theta_{12} = 0.2$.

2.2 Random Graph Models and Sparsity

So far, we have described *dense* Bernoulli graphs. For such models, as the sample size n increases, the expected degree of each node grows linearly. For example, consider the SBM with the third sampling setup described in section 2.1. Then the expected degree of vertex v_i is

$$\begin{aligned} E[d_i] &= \sum_j P_{ij} \\ &= \sum_{j \neq i} \theta_{z_i, z_j} \\ &= \Big(\sum_k n_k \theta_{z_i, k} \Big) - \theta_{z_i, z_i} \\ &= n \Big(\sum_k \alpha_k \theta_{z_i, k} \Big) - \theta_{z_i, z_i} \\ &= O(n). \end{aligned}$$

In many real networks, the degree of each vertex often does not grow proportionally with the size of the network. To account for this, a sparsity factor $\rho_n \in (0, 1]$ is introduced in the edge probabilities, i.e., $P_{ij} \leftarrow \rho_n P_{ij}$, for some sequence $\{\rho_n\}$ such that $\lim_{n \to \infty} \rho_n = 0$.

For example, a sparse SBM has edge probabilities $P_{ij} = \rho_n \theta_{z_i, z_j}$, for which we use the notation $A \sim \text{SBM}(z, \{\theta_{k\ell}\}_K; \rho_n)$ or $A \sim \text{SBM}(\alpha, \{\theta_{k\ell}\}_K; \rho_n)$, depending on whether we treat the labels as random or fixed. Then the expected degree grows as $O(n\rho_n)$ instead of linearly as $O(n)$. For the sake of unifying the sparse and dense regimes, we also allow for the special case $\rho_n = 1$ and include the sparsity factor throughout, unless otherwise stated. Finally, we also note that while ρ_n limits the rate of growth of the expected degree, our theoretical results still require $n\rho_n$ to diverge to infinity, albeit at a slower rate than $O(n)$ (see Abbe (2018) and Rubin-Delanchy et al. (2017) for further discussion).

2.3 Generalizations of the Stochastic Block Model: The Degree Corrected Block Model and the Popularity Adjusted Block Model

Definition 2.2 (Degree corrected block model)**.** Let $G = (V, E)$ be a Bernoulli graph with n vertices, described by random adjacency matrix A. Let $K \geq 1$ be an integer and $\theta_{k\ell} \in [0, 1]$ for each $k, \ell \in \{1, ..., K\}$ (if G is undirected, then $\theta_{k\ell} = \theta_{\ell k}$), as in the SBM. Let $z_1, ..., z_n \in \{1, ..., K\}$

be community labels associated with each vertex. In addition, each vertex v_i has a degree correction parameter $\omega_i \in [0, 1]$. If the edge probability matrix P for this graph is such that $P_{ij} = \rho_n \theta_{z_i, z_j} \omega_i \omega_j$ and $A \sim \text{BernoulliGraph}(P)$, then G is a degree corrected block model. We use the notation $A \sim \text{DCBM}(z, \{\theta_{k\ell}\}_K, \omega; \rho_n)$ to denote a random adjacency matrix drawn from the DCBM with labels z, base edge probabilities $\{\theta_{k\ell}\}_K$, and degree correction factors ω. Under the sampling scheme in which the labels are drawn from a multivariate distribution with class probabilities α, we use the notation $A \sim \text{DCBM}(\alpha, \{\theta_{k\ell}\}_K, \omega; \rho_n)$.

Example 2.4 (Dating network as a disassortative DCBM)**.** In example 2.2, we modeled an online dating network as an SBM in which the vertices correspond to individuals, the communities correspond to genders, and edges correspond to whether the online dating service successfully matched pairs of individuals. In this model, we assume that for each pair of genders, there is a constant probability of a match. This does not take into account that some individuals are more likely to form connections than others, based on each individual's activity on the online service or popularity with other users. Modeling this as a DCBM allows us to take this into account by adjusting each user's degree by ω_i. The following is a visualization of the network from example 2.2 modified as a DCBM with parameters $\theta_{11} = \theta_{22} = 0.03$, $\theta_{12} = 0.3$, each ω_i drawn uniformly between $1/3$ and 1, and $\rho_n \equiv 1$.

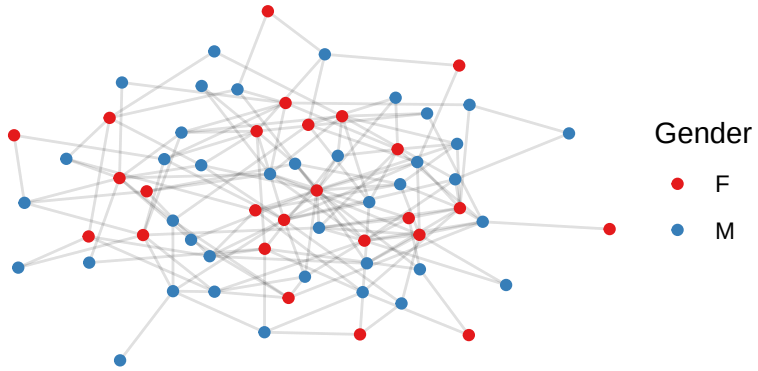

Figure 4: A dating network modeled as a DCBM. This model is disassortative.

Definition 2.3 (Popularity adjusted block model)**.** Let $G = (V, E)$ be an undirected Bernoulli graph with n vertices, described by random adjacency matrix A. Let $K \geq 1$ be an integer that describes the number of communities, and let $z_1, ..., z_n \in \{1, ..., K\}$ be community labels associated

with each vertex. Suppose that each vertex v_i has K popularity parameters $\lambda_{i1}, ..., \lambda_{iK}$ for which each λ_{ik} describes v_i's affinity toward community k. Then if the edge probability matrix P for this graph is such that $P_{ij} = \rho_n \lambda_{i,z_j} \lambda_{j,z_i}$ and $A \sim \text{BernoulliGraph}(P)$, then G is a popularity adjusted block model. We use the notation $A \sim \text{PABM}(z, \{\lambda_{i,k}\}_K; \rho_n)$ to denote a random adjacency matrix drawn from the PABM with labels z and popularity parameters $\{\lambda_{i,k}\}_K$. Under the sampling scheme in which the labels are drawn from a multivariate distribution with class probabilities α, we use the notation $A \sim \text{PABM}(\alpha, \{\lambda_{i,k}\}_K; \rho_n)$.

Remark. Unlike the SBM (when excluding the sparsity factor), each DCBM is not unique and therefore non-identifiable. For example, doubling each $\theta_{k\ell}$ and dividing each ω_i by $\sqrt{2}$ results in the same edge probability matrix. Nevertheless, a naive EM type algorithm similar to the one in section 2.1 results in high community detection accuracy for most networks, although it is not clear what its asymptotic properties are or whether there are any theoretical guarantees. Likewise, it is not clear

Example 2.5 (Dating network as a PABM)**.**

2.4 The Hierarchy of Block Models

3 Random Dot Product Graphs and Generalized Random Dot Product Graphs

3.1 Definitions

Definition 3.1 (Generalized random dot product graph)**.** Let $p \geq 1$ and $q \geq 0$ be integers. Define $I_{p,q}$ as the block diagonal matrix $I_{p,q} = \left[\begin{smallmatrix} I_p & 0 \\ 0 & -I_q \end{smallmatrix}\right]$ where I_p and I_q are the identity matrices of dimensions $p \times p$ and $q \times q$ respectively. Denote $d = p + q$ and let $\mathcal{X}$ be the subset of $\mathbb{R}^d$ such that, for any $x, y \in \mathcal{X}$, we have $x^\top I_{p,q} y \in [0, 1]$. Let $X = \begin{bmatrix} x_1 & \dots & x_n \end{bmatrix}^\top$ be an $n \times d$ matrix with rows $x_i \in \mathcal{X}$. A graph G with adjacency matrix A is said to be a generalized random dot product graph with latent positions X, sparsity parameter ρ_n, and signature (p, q) if $A \sim \text{BernoulliGraph}(P)$ where the edge probability matrix P is given by $P = \rho_n X I_{p,q} X^\top$, i.e., the entries of P are of the form $P_{ij} = \rho_n x_i^\top I_{p,q} x_j$.

We use the notation $A \sim \text{GRDPG}_{p,q}(X; \rho_n)$ to denote a random adjacency matrix A drawn from latent positions X, sparsity parameter ρ_n, and signature (p, q). If the n vectors in X are drawn from probability distribution F on support $\mathcal{X}$, we use the notation $A \sim \text{GRDPG}_{p,q}(F, n; \rho_n)$.

If $q = 0$ and so $d = p$, (i.e., P is positive semidefinite), then we call this a random dot product graph. In this case, we use the notation $A \sim \text{RDPG}(X; \rho_n)$ and $A \sim \text{RDPG}(F, n; \rho_n)$.

Definition 3.2 (Adjacency spectral embedding)**.**

Definition 3.3 (Indefinite orthogonal group)**.**

Remark. The latent vectors that produce $X I_{p,q} X^\top = P$ are not unique (Rubin-Delanchy et al., 2017). More specifically, if $P_{ij} = x_i^\top I_{p,q} x_j$, then for any $Q \in \mathbb{O}(p, q)$ we also have $(Qx_i)^\top I_{p,q} (Qx_j) = x_i^\top (Q^\top I_{p,q} Q) x_j = x_i^\top I_{p,q} x_j = P_{ij}$. Unlike in the RDPG case, transforming the latent positions via multiplication by $Q \in \mathbb{O}(p, q)$ does not necessarily maintain interpoint angles or distances.

3.2 Connecting the SBM and DCBM to the GRDPG

4 Popularity Adjusted Block Models are Generalized Random Dot Product Graphs

In this section, we connect the PABM to the GRDPG and exploit that connection to develop algorithms for community detection and parameter estimation. In order to make the explicit connection between the PABM and the GRDPG, we make use of an alternative but equivalent definition of the PABM which parameterizes the model in terms of popularity vectors, which are collections of popularity parameters.

Remark. In a PABM, each vertex i has K popularity parameters $\lambda_{i1}, \ldots, \lambda_{iK}$, that describe its affinity toward each of the K communities. Another view of a PABM is as follows. Let $\tilde{P}$ be the matrix obtained by permuting the rows and columns of P so that the vertices are reorganized by community memberships $z_i \in \{1, 2, \ldots, K\}$ in increasing order. Denote by $\tilde{P}^{(k\ell)}$ the $n_k \times n_\ell$ submatrix of $\tilde{P}$ corresponding to the edge probabilities between vertices in communities k and ℓ. Note that $\tilde{P}^{(k\ell)} = (\tilde{P}^{(\ell k)})^\top$. Next let $\lambda^{(k\ell)} = \{\lambda_{i\ell} \colon z_i = k\} \in \mathbb{R}^{n_k}$; the elements of $\lambda^{(k\ell)}$ are the affinity parameters toward the ℓth community of all vertices in the k^{th} community. Define $\lambda^{(\ell k)}$ analogously. Then each block $\tilde{P}^{(k\ell)}$ can be written as the outer product of two vectors:

$$\tilde{P}^{(k\ell)} = \rho_n \lambda^{(k\ell)} (\lambda^{(\ell k)})^\top. \tag{4}$$

We will henceforth use the notation $A \sim \text{PABM}(\{\lambda^{(k\ell)}\}_K, \rho_n)$ to denote a random adjacency matrix A drawn from a PABM with K communities, popularity parameters $\{\lambda^{(k\ell)}\}$ and sparsity parameter ρ_n.

4.1 The Geometry of PABMs

In section 3.2, we showed how the SBM and DCBM are special cases of the GRDPG in which the latent vectors form a very particular geometry. Similarly, we now show the special geometry of the PABM when viewed as a GRDPG. For ease of exposition, and without loss of generality, we drop the dependency on the sparsity parameter ρ_n and assume $\rho_n \equiv 1$ throughout this subsection.

Theorem 4.1 (The latent configuration of the PABM)**.** *Let $A \sim \text{PABM}(\{\lambda^{(k\ell)}\}_K)$ be an instance*

of a PABM with $K \geq 1$ blocks and latent vectors $\{\lambda^{(k\ell)} : 1 \leq k \leq K, 1 \leq \ell \leq K\}$. Then there exists a block diagonal matrix $X \in \mathbb{R}^{n \times K^2}$ defined by $\{\lambda^{(k\ell)}\}$ and a $K^2 \times K^2$ fixed orthonormal matrix U such that $A \sim \mathrm{GRDPG}_{K(K+1)/2, K(K-1)/2}(\tilde{\Pi} X U)$. Here $\tilde{\Pi}$ is the permutation matrix such that $P = \tilde{\Pi}\tilde{P}\tilde{\Pi}^\top$ where the rows and columns of $\tilde{P}$ are arranged according to increasing values of the community labels (see remark 4).

Proof. We will prove this theorem in two parts. First, for demonstration purposes, we focus on the case when $K = 2$ to build intuition. The general case of $K \geq 2$ is presented later.

For the $K = 2$ case, the proof is straightforward. We will first work with the matrix $\tilde{P}$. Note that $\tilde{P}$ has the form

$$\tilde{P} = \begin{bmatrix} P^{(11)} & P^{(12)} \\ P^{(21)} & P^{(22)} \end{bmatrix} = \begin{bmatrix} \lambda^{(11)}(\lambda^{(11)})^\top & \lambda^{(12)}(\lambda^{(21)})^\top \\ \lambda^{(21)}(\lambda^{(12)})^\top & \lambda^{(22)}(\lambda^{(22)})^\top \end{bmatrix}.$$

Now let

$$X = \begin{bmatrix} \lambda^{(11)} & \lambda^{(12)} & 0 & 0 \\ 0 & 0 & \lambda^{(21)} & \lambda^{(22)} \end{bmatrix} \quad \text{and} \quad U = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1/\sqrt{2} & 1/\sqrt{2} \\ 0 & 0 & 1/\sqrt{2} & -1/\sqrt{2} \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Then by straightforward matrix multiplication, we obtain

$$XUI_{3,1}U^\top X^\top = \begin{bmatrix} \lambda^{(11)}(\lambda^{(11)})^\top & \lambda^{(12)}(\lambda^{(21)})^\top \\ \lambda^{(21)}(\lambda^{(12)})^\top & \lambda^{(22)}(\lambda^{(22)})^\top \end{bmatrix} = \tilde{P}$$

and hence $\tilde{P}$ also corresponds to the edge probability matrix of GRDPG with latent vectors described by XU. As $P = \tilde{\Pi}\tilde{P}\tilde{\Pi}^\top$ we conclude that P has latent vectors described by $\tilde{\Pi}XU$.

It is nevertheless instructive to look at a few intermediate steps. More specifically, the product $UI_{3,1}U^\top$ yields a permutation matrix Π with fixed points at positions 1 and 4 and a cycle of order

2 swapping positions 2 and 3, i.e.,

$$\Pi = UI_{3,1}U^\top = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Furthermore, as U is orthonormal and $I_{3,1}$ is diagonal, $UI_{3,1}U^\top$ is also an eigendecomposition of Π where the fixed points of Π are mapped to the eigenvectors e_1 and e_4 while the cycles of order two are mapped to the eigenvectors $\frac{1}{\sqrt{2}}(e_2 + e_3)$ and $\frac{1}{\sqrt{2}}(e_2 - e_3)$; here e_i denote the i^{th} basis vector in $\mathbb{R}^4$. Thus, another way of decomposing the edge probability matrix is $\tilde{P} = X\Pi X^\top$ where the rows of X lie in the union of two 2-dimensional orthogonal subspaces and Π is a permutation matrix.

For the general case, we can extend $\tilde{P} = X\Pi X^\top$ to larger K. For a more concrete example of this, refer to Example 1. We once again consider $\tilde{P}$ as defined in remark 4. We first define the following matrices

$$\Lambda^{(k)} = \left[\lambda^{(k1)} \mid \cdots \mid \lambda^{(kK)}\right] \in \mathbb{R}^{n_k \times K}, \quad X = \text{blockdiag}(\Lambda^{(1)}, \ldots, \Lambda^{(K)}) \in \mathbb{R}^{n \times K^2}, \tag{5}$$

$$L^{(k)} = \text{blockdiag}(\lambda^{(1k)}, \ldots, \lambda^{(Kk)}) \in \mathbb{R}^{n \times K}, \quad Y = \left[L^{(1)} \mid \cdots \mid L^{(K)}\right] \in \mathbb{R}^{n \times K^2}. \tag{6}$$

It is then straightforward to verify that

$$XY^\top = \text{blockdiag}(\Lambda^{(1)}, \ldots, \Lambda^{(K)}) \begin{bmatrix} L_1^\top \\ \vdots \\ L_K^\top \end{bmatrix} = \begin{bmatrix} \Lambda^{(1)}(L^{(1)})^\top \\ \vdots \\ \Lambda^{(K)}(L^{(K)})^\top \end{bmatrix},$$

$$\Lambda^{(k)}(L^{(k)})^\top = \left[\lambda^{(k1)}(\lambda^{(1k)})^\top \mid \cdots \mid \lambda^{(kK)}(\lambda^{(Kk)})^\top\right] = \left[P^{(k1)} \mid P^{(k2)} \mid \cdots \mid P^{(kK)}\right].$$

We therefore have $\tilde{P} = XY^\top$. Similar to the $K = 2$ case, we also have $Y = X\Pi$ for some permutation matrix Π and hence $\tilde{P} = X\Pi X^\top$. The permutation described by Π now has K fixed

points, which correspond to K eigenvalues equal to 1 with corresponding eigenvectors e_k where $k = r(K+1)+1$ for $0 \leq r \leq K-1$. It also has $\binom{K}{2}$ cycles of order 2. Each cycle corresponds to a pair of eigenvalues $\{-1, +1\}$ and a pair of eigenvectors $\{(e_s + e_t)/\sqrt{2}, (e_s - e_t)/\sqrt{2}\}$.

Let $p = K(K+1)/2$ and $q = K(K-1)/2$. We therefore have

$$\Pi = U I_{p,q} U^\top \tag{7}$$

where U is a $K^2 \times K^2$ orthogonal matrix and hence

$$\tilde{P} = XU I_{p,q} (XU)^\top. \tag{8}$$

In summary we can describe the PABM with K communities as a GRDPG with latent positions $\tilde{\Pi} X U$ and known signature $(p, q) = (\frac{1}{2}K(K+1), \frac{1}{2}K(K-1))$. □

Example 4.1. Let A be a 3 blocks PABM with latent vectors $\{\lambda^{(k\ell)} : 1 \leq k \leq 3, 1 \leq \ell \leq 3\}$. Using the same notation as in theorem 4.1, we can define

$$X = \begin{bmatrix} \lambda^{(11)} & \lambda^{(12)} & \lambda^{(13)} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda^{(21)} & \lambda^{(22)} & \lambda^{(23)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda^{(31)} & \lambda^{(32)} & \lambda^{(33)} \end{bmatrix},$$

$$Y = \begin{bmatrix} \lambda^{(11)} & 0 & 0 & \lambda^{(12)} & 0 & 0 & \lambda^{(13)} & 0 & 0 \\ 0 & \lambda^{(21)} & 0 & 0 & \lambda^{(22)} & 0 & 0 & \lambda^{(23)} & 0 \\ 0 & 0 & \lambda^{(31)} & 0 & 0 & \lambda^{(32)} & 0 & 0 & \lambda^{(33)} \end{bmatrix}.$$

Then $Y = X\Pi$ and $\tilde{P} = XY^\top$ where Π is a 9×9 permutation matrix of the form

$$\Pi = \Big[e_1 \mid e_4 \mid e_7 \mid e_2 \mid e_5 \mid e_8 \mid e_3 \mid e_6 \mid e_9 \Big].$$

where e_i denotes the i^{th} basis vector in $\mathbb{R}^9$. The matrix Π corresponds to a permutation of $\{1, 2, \ldots, 9\}$ with the following decomposition.

1. Positions 1, 5, 9 are fixed.

2. There are three cycles of length 2, namely $(2,4)$, $(3,7)$, and $(6,8)$.

We can thus write Π as $\Pi = UI_{6,3}U^\top$ where the first three columns of U consist of e_1, e_5, and e_9 corresponding to the fixed points, the next three columns are the eigenvectors $(e_k + e_\ell)/\sqrt{2}$, and the last three columns are the eigenvectors $(e_k - e_\ell)/\sqrt{2}$ for $(k,\ell) \in \{(2,4),(3,7),(6,8)\}$.

The matrix $\tilde{P}$ is then the edge probabilities matrix for a Generalized Random Dot Product Graph whose latent positions are the rows of the matrix

$$XU = \begin{bmatrix} \lambda^{(11)} & 0 & 0 & \frac{\lambda^{(12)}}{\sqrt{2}} & \frac{\lambda^{(13)}}{\sqrt{2}} & 0 & \frac{\lambda^{(12)}}{\sqrt{2}} & \frac{\lambda^{(13)}}{\sqrt{2}} & 0 \\ 0 & \lambda^{(22)} & 0 & \frac{\lambda^{(21)}}{\sqrt{2}} & 0 & \frac{\lambda^{(23)}}{\sqrt{2}} & -\frac{\lambda^{(21)}}{\sqrt{2}} & 0 & \frac{\lambda^{(23)}}{\sqrt{2}} \\ 0 & 0 & \lambda^{(33)} & 0 & \frac{\lambda^{(31)}}{\sqrt{2}} & \frac{\lambda^{(32)}}{\sqrt{2}} & 0 & -\frac{\lambda^{(31)}}{\sqrt{2}} & -\frac{\lambda^{(32)}}{\sqrt{2}} \end{bmatrix}$$

and the latent positions for P is a permutation of the rows of XU.

4.2 Algorithms

Two inference objectives arise from the PABM:

1. Community membership identification (up to permutation).
2. Parameter estimation (estimating λ_{ik}'s).

In our methods, the data that are observed for estimation is the adjacency matrix, $A \sim \text{PABM}(\{\lambda^{(k\ell)}\}_K, \rho_n)$, along with an assumed number of communities, K. To motivate our methods, we first consider community detection and parameter estimation in the case where we know the edge probability matrix P beforehand, noting that community memberships and popularity parameters are not immediately discernible from P itself. After establishing methods for community detection and parameter estimation from P, we use the consistency property of the ASE (Sussman et al., 2012, Rubin-Delanchy et al., 2017) to demonstrate that the same methods work for A almost surely as $n \to \infty$.

4.2.1 Previous Work

Sengupta and Chen (2018) used Modularity Maximization (MM) and the Extreme Points (EP) algorithm (Le et al., 2016) for community detection and parameter estimation. They were able to

show that as the sample size increases, the *proportion* of misclassified community labels (up to permutation) goes to 0.

Noroozi et al. (2019) used Sparse Subspace Clustering (SSC) (Elhamifar and Vidal, 2009) for community detection in the PABM. The SSC algorithm can be described as follows: Given $X \in \mathbb{R}^{n\times d}$ with vectors $x_i^\top \in \mathbb{R}^d$ as rows of X, the optimization problem $c_i = \text{argmin}_c \|c\|_1$ subject to $x_i = X^\top c$ and $c^{(i)} = 0$, where $c^{(i)}$ is the i^{th} entry of c, is solved for each $i \in [n]$. The solutions are collected into matrix $C = \left[c_1 \mid \cdots \mid c_n\right]^\top$ to construct an affinity matrix $B = |C| + |C^\top|$. If each x_i lies exactly on one of K subspaces, B describes an undirected graph consisting of *at least* K disjoint subgraphs, i.e., $B_{ij} = 0$ if x_i, x_j lie on different subspaces. The intuition here is that vectors that lie on the same subspace can be described as linear combinations of each other, assuming the number of vectors in the subspace is greater than the dimensionality of the subspace. Then once sparsity is enforced, for each c_i, its j^{th} element $c_i^{(j)}$ is zero if x_j belongs to a subspace that doesn't contain x_i, resulting in $B_{ij} = 0$. If X instead represents points near K subspaces with some noise, then this property may only hold approximately and a final graph partitioning step may be required (e.g., edge thresholding or spectral clustering).

In practice, due to presence of noise, SSC is often done by solving the LASSO problems

$$c_i = \text{argmin}_c \frac{1}{2}\|x_i - X_{-i}^\top c\|_2^2 + \vartheta\|c\|_1 \tag{9}$$

for some sparsity parameter $\vartheta > 0$. The c_i vectors are then collected into C and B as before.

Definition 4.1 (Subspace Detection Property)**.** Let $X = \left[x_1 \mid \cdots \mid x_n\right]^\top$ be noisy points sampled from K subspaces, i.e., $x_i = y_i + z_i$ where the y_i belongs to the union of K subspaces and the z_i are noise vectors. Let $\vartheta \geq 0$ be given and let C and B be constructed from the solutions of LASSO problems as described in equation (9) with this given choice of ϑ. Then X is said to satisfy the subspace detection property with sparsity parameter ϑ if each column of C has nonzero ℓ_2 norm and $B_{ij} = 0$ whenever y_i and y_j are from different subspaces.

Remark. One common approach to show that SSC works for a noisy sample X is to show that X satisfies the subspace detection property for some choice of ϑ; recall that ϑ is the sparsity parameter for the LASSO problems in equation (9). However, this is not sufficient to guarantee

that SSC perfectly recovers the underlying subspaces. More specifically, if X satisfies the subspace detection property, then B describes a graph with *at least* K disconnected subgraphs, with the ideal case being that there are exactly K subgraphs which map onto each subspace. Nevertheless it is also possible that the K subspaces are represented by $K' > K$ multiple disconnected subgraphs and we cannot, at least without a subsequent post-processing step, recover the K subspaces directly from B; see Nasihatkon and Hartley (2011) and Liu et al. (2013) for further discussions. Therefore in practice B is usually treated as an affinity matrix and, as we allude to earlier, the rows of B are partitioned using some clustering algorithm to obtain the final clustering.

Theorem 4.1 suggests that SSC is appropriate for community detection for the PABM, provided that we observe the edge probabilities matrix P. More precisely, given the matrix $\tilde{P}$ obtained by permuting the rows and columns of P as described in remark 4 we can recover XU up to some non-identifiability indefinite orthogonal transformation Q. Then using results from Soltanolkotabi and Candés (2012), it can be easily shown that the subspace detection property holds for XU. Indeed, the columns of XU from different communities correspond to mutually orthogonal subspaces. This then implies that the subspace detection property also holds for XUQ for all invertible transformation Q and hence the subspace detection property also holds for $\tilde{\Pi}XUQ$ for any $n \times n$ permutation matrix $\tilde{\Pi}$.

However, because we do not observe P but rather only the noisy adjacency matrix $A \sim \text{BernoulliGraph}(P)$, the natural approach then is to perform SSC on the rows of the spectral embedding of A, since the embedding of P consists of K subspaces (theore 4.1), and so the embedding of A will lie approximately on the K subspaces. We will show in theorem 4.4 that, with probability converging to one as $n \to \infty$, the rows of the ASE of A also satisfy the subspace detection property. Theorem 4.4 builds upon existing work by Rubin-Delanchy et al. (2017) who describe the convergence behavior of the ASE of A to that of $\tilde{\Pi}XU$, and Wang and Xu (2016) who show the necessary conditions for the subspace detection property to hold in noisy cases where the points lie near subspaces. Finally we emphasize that while Noroozi et al. (2019) also considered the use of SSC for community recovery in PABM, they instead applied SSC to the rows of A itself, foregoing the embedding step altogether. It is however much harder to show that the rows of A satisfy the subspace detection property and thus, to the best of our knowledge, there is currently no

consistency result regarding the application of SSC to the rows of A.

4.2.2 Algorithms for Community Detection

We previously stated in theorem 4.1 one possible set of latent positions that result in the edge probability matrix of a PABM, namely $P = \tilde{\Pi}(XU)I_{p,q}(XU)^\top\tilde{\Pi}^\top$ where X is block diagonal and $\tilde{\Pi}$ is a permutation matrix.

Furthermore, the explicit form of XU represents points in $\mathbb{R}^{K^2}$ such that points within each community lie on K-dimensional orthogonal subspaces, i.e. $\langle U^\top x_i, U^\top x_j\rangle = 0$ whenever i and j are in different communities. Thus if we have (or can estimate) XU directly, then both the community detection and parameter identification problem are trivial because U is orthonormal and fixed for each value of K. However, direct identification or estimation of XU is possibly difficult due to the non-identifiability of XU (see remark 3.1) when we are given only P. More specifically, suppose we find a matrix $Y \in \mathbb{R}^{n\times K^2}$ such that $P = YI_{p,q}Y^\top$. Then it is generally the case that $Y = \tilde{\Pi}XUQ$ for some indefinite orthogonal matrix $Q \in \mathbb{O}(p,q)$. However since Q is not necessarily an orthogonal matrix and hence, if y_i denote the i^{th} row of Y, then $\langle U^\top x_i, U^\top x_j\rangle \neq \langle y_i, y_j\rangle$. This prevents us from transferring the orthogonality property of XU directly to Y.

Nevertheless by using the special geometric structure of X we can circumvent the non-identifiability of Y and XU by using instead the rows of the matrix V of eigenvectors (corresponding to the non-zero eigenvalues) of P. In particular V is identifiable up to orthogonal transformations and furthermore, due to the block diagonal structure of X, the rows of V also lie on K distinct orthogonal subspaces and hence $v_i^\top v_j = 0$ whenever $z_i \neq z_j$.

Theorem 4.2. *Let $P = VDV^\top$ be the spectral decomposition of the edge probability matrix. Let $B = nVV^\top$. Assume $\lambda_{iz_i} > 0$ for each $i \in [n]$, i.e., each vertex's popularity parameter to its own community is nonzero. Then $B_{ij} = 0$ if and only if vertices i and j are in different communities.*

Proof. We first show that $VV^\top = \tilde{\Pi}X(X^\top X)^{-1}X^\top\tilde{\Pi}^\top$ where X is defined as in Eq. (5). Indeed, by Theorem 2, $P = \tilde{\Pi}XUI_{p,q}U^\top X^\top\tilde{\Pi}$ for $p = K(K+1)/2$ and $q = K(K-1)/2$. The eigendecomposition $P = VDV^\top$ also yields $P = V|D|^{1/2}I_{p,q}|D|^{1/2}V^\top$ where $|\cdot|^{1/2}$ is applied entry-wise. Now let $Y = \tilde{\Pi}XU$ and $\tilde{Y} = V|D|^{1/2}$; note that Y and $\tilde{Y}$ both have full column ranks.

Because $P = YI_{p,q}Y^\top = \tilde{Y}I_{p,q}\tilde{Y}^\top$, we have

$$Y = \tilde{Y}I_{p,q}\tilde{Y}^\top Y(Y^\top Y)^{-1}I_{p,q}.$$

Let $Q = I_{p,q}\tilde{Y}^\top Y(Y^\top Y)^{-1}I_{p,q}$ and note that $Y = \tilde{Y}Q$. We then have

$$\begin{aligned} Q^\top I_{p,q}Q &= I_{p,q}(Y^\top Y)^{-1}Y^\top \tilde{Y}I_{p,q}I_{p,q}I_{p,q}\tilde{Y}^\top Y(Y^\top Y)^{-1}I_{p,q} \\ &= I_{p,q}(Y^\top Y)^{-1}Y^\top Y I_{p,q}Y^\top Y(Y^\top Y)^{-1}I_{p,q} = I_{p,q} \end{aligned}$$

and hence Q is an indefinite orthogonal matrix.

Let $R = UQ|D|^{-1/2}$ and note that $V = \tilde{\Pi}XR$. Because R is invertible, we can write

$$\tilde{\Pi}X(X^\top X)^{-1}X^\top \tilde{\Pi}^\top = \tilde{\Pi}XR(R^\top X^\top XR)^{-1}R^\top X^\top \tilde{\Pi}^\top.$$

Furthermore, as V has orthonormal columns, $R^\top X^\top XR = V^\top \tilde{\Pi}\tilde{\Pi}^\top V = V^\top V = I$. We thus conclude

$$\tilde{\Pi}X(X^\top X)^{-1}X^\top \tilde{\Pi}^\top = V(V^\top V)^{-1}V^\top = VV^\top$$

as desired.

To complete the proof of theorem 4.2, recall that X is block diagonal with each block corresponding to one community, and hence $X(X^\top X)^{-1}X^\top$ is also a block diagonal matrix with each block corresponding to a community. As $B = nVV^\top = n\tilde{\Pi}X(X^\top X)^{-1}X^\top \tilde{\Pi}^\top$, we conclude that $B_{ij} = 0$ whenever vertices i and j belong to different communities. □

Theorem 4.2 provides perfect community detection from P. More specifically, let $|B|$ be the affinity matrix for graph G', where $|\cdot|$ is applied entry-wise. Then G' consists of exactly K disjoint subgraphs, as G' has no edges between communities. All that is left to identify the communities is to assign each subgraph a distinct community label. In practice, we do not observe P and instead only observe the noisy $A \sim \text{BernoulliGraph}(P)$. A natural approach is then to use the affinity matrix $\hat{B} = n\hat{V}\hat{V}^\top$ where $\hat{V}$ is the matrix of eigenvectors (corresponding to the largest eigenvalues in modulus) of A. The resulting procedure, named Orthogonal Spectral Clustering, is presented in

Algorithm 2: Orthogonal Spectral Clustering.

Data: Adjacency matrix A, number of communities K

Result: Community assignments $z_1, ..., z_n \in \{1, ..., K\}$

1 Compute the eigenvectors of A that correspond to the $K(K+1)/2$ most positive eigenvalues and $K(K-1)/2$ most negative eigenvalues. Construct V using these eigenvectors as its columns.

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 (e.g., using edge thresholding or spectral clustering).

5 Map each partition to the community labels $1, ..., K$.

algorithm 2. The following result leverages existing theoretical properties of ASE for estimating of latent positions in a GRDPG (Rubin-Delanchy et al., 2017) to show that $\hat{B}$ converges almost surely to B; in particular $\hat{B}_{ij} \xrightarrow{a.s.} 0$ for each pair (i, j) in different communities.

Theorem 4.3. *Assume the setting of theorem 4.2. Let $\hat{B}$ with entries $\hat{B}_{ij}$ be the affinity matrix obtained from OSC as described in algorithm 2. Then for $n\rho_n = \omega(\log^4 n)$, we have*

$$\max_{i,j} |\hat{B}_{ij} - B_{ij}| = O\Big(\frac{\log n}{\sqrt{n\rho_n}}\Big) \tag{10}$$

with high probability. In particular $\hat{B}_{ij} - B_{ij} \xrightarrow{\text{a.s.}} 0$ where the convergence is uniform over all i, j. Hence for all pairs (i, j) in different communities we have $\hat{B}_{ij} \xrightarrow{\text{a.s.}} 0$, while for all pairs (i, j) in the same community, $\liminf_{n\to\infty} |\hat{B}_{ij}| > 0$ almost surely.

Theorem 4.3 guarantees that for any $\epsilon > 0$, the number of edges of $\hat{B}$ between vertices of different communities that are larger than ϵ converges to zero with probability converging to one as n increases. We can always find an $\epsilon > 0$ such that $\hat{B}_{ij} > \epsilon$ with probability converging to one as n increases. Thus, by using $\hat{B}$, we can perfectly recover all the latent community assignments $z_1, z_2, \ldots, z_n$, i.e., the number of misclustered vertices is zero asymptotically almost surely. We note that theorem 4.3 is stronger than existing results in the literature; in particular theorem 1 of Sengupta and Chen (2018) (the paper that originally introduces the PABM model) only guarantees that the *proportion* of misclustered vertices converges to 0 as $n \to \infty$. Furthermore theorem 1 of Sengupta and Chen (2018) also requires the sparsity parameter ρ_n to satisfies $n\rho_n^2 = \omega(\log^2 n)$

Algorithm 3: Sparse Subspace Clustering using LASSO.

Data: Adjacency matrix A, number of communities K, hyperparameter λ

Result: Community assignments $z_1, ..., z_n \in \{1, ..., K\}$

Find V, the matrix of eigenvectors of A corresponding to the $K(K+1)/2$ most positive and the $K(K-1)/2$ most negative eigenvalues.

Normalize $V \leftarrow \sqrt{n}V$.

for $i = 1, ..., n$ **do**

  Assign $v_i^\top$ as the i^{th} row of V. Assign $V_{-i} = \left[v_1 \mid \cdots \mid v_{i-1} \mid v_{i+1} \mid \cdots \mid v_n\right]^\top$.

  Solve the LASSO problem $c_i = \arg\min_\beta \frac{1}{2}\|v_i - V_{-i}\beta\|_2^2 + \lambda\|\beta\|_1$.

  Assign $\tilde{c}_i = (c_i^{(1)}, \ldots, c_i^{(i-1)}, 0, c_i^{(i)}, \ldots, c_i^{(n-1)})^\top$ such that the superscript is the index of $\tilde{c}_i$.

end

Assign $C = \left[\tilde{c}_1 \mid \cdots \mid \tilde{c}_n\right]$.

Compute the affinity matrix $B = |C| + |C^\top|$.

Construct graph G using B as its similarity matrix.

Partition G into K disconnected subgraphs (e.g., using edge thresholding or spectral clustering).

Map each partition to the community labels $1, ..., K$.

which is a considerably stronger assumption than the assumption $n\rho_n = \omega(\log^4 n)$ used in theorem 4.3. Indeed, $n\rho_n^2 = \omega(\log^2 n)$ implies $n\rho_n = \omega(n^{1/2})$. We emphasize that the assumption $n\rho_n = \omega(\log^c n)$ for some constant $c > 1$ is commonly used in the context of graph estimation using spectral methods.

Theorems 4.1, 4.2, and 4.3 also provide a natural path toward using SSC for community detection. In particular we established in theorem 4.1 that an ASE of the edge probability matrix P can be constructed from a latent vector configuration consisting of orthogonal subspaces. Theorem 4.2 shows how this property can also be recovered from the eigenvectors of P. Then theorem 4.3 shows that, by replacing P with A, the rows of $\hat{V}$ also lie on asymptotically orthogonal subspaces. Motivated by theorem 4.3, theorem 4.4 below shows that the subspace detection property also holds for the rows of $\sqrt{n}\hat{V}$.

Theorem 4.4. *Let P describe the edge probability matrix of the PABM with n vertices, and let $A \sim \text{Bernoulli}(P)$. Let $\hat{V}$ be the matrix of eigenvectors of A corresponding to the K^2 largest*

Algorithm 4: PABM parameter estimation.

Data: Adjacency matrix A, community assignments $1, ..., K$

Result: PABM parameter estimates $\{\hat{\lambda}^{(k\ell)}\}_K$.

Arrange the rows and columns of A by community such that each $A^{(k\ell)}$ block consists of estimated edge probabilities between communities k and l.

for $k, \ell = 1, ..., K,\ k \leq \ell$ **do**

  Compute $A^{(k\ell)} = U\Sigma V^\top$, the SVD of the $k\ell^{th}$ block.

  Assign $u^{(k\ell)}$ and $v^{(k\ell)}$ as the first columns of U and V. Assign $(\sigma^{(k\ell)})^2 \leftarrow \Sigma_{11}$.

  Assign $\hat{\lambda}^{(k\ell)} \leftarrow \pm\sigma^{(k\ell)}u^{(k\ell)}$ and $\hat{\lambda}^{(\ell k)} \leftarrow \pm\sigma^{(k\ell)}v^{(k\ell)}$.

end

eigenvalues in modulus. Then for any $\epsilon > 0$, there exists a choice of $\vartheta > 0$ and $N \in \mathbb{N}$ such that for all $n \geq N$, $\sqrt{n}\hat{V}$ obeys the subspace detection property with probability at least $1 - \epsilon$.

4.2.3 Algorithm for Parameter Estimation

For ease of exposition we now assume in this subsection that the edge probability matrix P for the PABM had been arranged so that the rows and columns are organized by community so that $\tilde{P} = P$ (see remark 4). Then the $k\ell^{\text{th}}$ block is an outer product of two vectors, i.e., $P^{(k\ell)} = \lambda^{(k\ell)}(\lambda^{(\ell k)})^\top$. Therefore, given $P^{(k\ell)}$, $\lambda^{(k\ell)}$ and $\lambda^{(\ell k)}$ are solvable up to multiplicative constant using singular value decomposition. More specifically, let $P^{(k\ell)} = (\sigma^{(k\ell)})^2 u^{(k\ell)}(v^{(k\ell)})^\top$ be the singular value decomposition of $P^{(k\ell)}$ where $u^{(k\ell)} \in \mathbb{R}^{n_k}$ and $v^{(k\ell)} \in \mathbb{R}^{n_\ell}$ are vectors and $\sigma^{(k\ell)}$ is a scalar. Then $\rho_n^{1/2}\lambda^{(k\ell)} = s_1 u^{(k\ell)}$ and $\rho_n^{1/2}\lambda^{(\ell k)} = s_2 v^{(k\ell)}$ for unidentifiable $s_1 s_2 = (\sigma^{(k\ell)})^2$. Because each $\lambda^{(k\ell)}$ is not strictly identifiable, we instead estimate each $\tilde{\lambda}^{(k\ell)} = \sigma^{(k\ell)}u^{(k\ell)}$. Given the adjacency matrix A instead of edge probability matrix P, we can simply use plug-in estimators by taking the SVD of each $A^{(k\ell)}$ to obtain $\hat{\lambda}^{(k\ell)} = \hat{\sigma}^{(k\ell)}\hat{u}^{(k\ell)}$ using the largest singular value of A and its corresponding singular vectors.

Theorem 4.5. *Let each $\tilde{\lambda}^{(k\ell)}$ be the popularity vector derived from its corresponding $P^{(k\ell)}$ and let $\hat{\lambda}^{(k\ell)}$ be its estimate obtained from $A^{(k\ell)}$ using algorithm 4. Then if $n\rho_n = \omega(\log^4 n)$,*

$$\max_{k,\ell \in \{1,...,K\}} \|\hat{\lambda}^{(k\ell)} - \tilde{\lambda}^{(k\ell)}\|_\infty = O\left(\frac{\log n_k}{\sqrt{n_k}}\right) \tag{11}$$

with high probability. Here $\|\cdot\|_\infty$ *denotes the* ℓ_∞ *norm of a vector. Let* $\hat{\Lambda}$ *be the matrix*

$$\hat{\Lambda} = \begin{bmatrix} \hat{\lambda}^{(11)} & \hat{\lambda}^{(12)} & \dots & \hat{\lambda}^{(1K)} \\ \hat{\lambda}^{(21)} & \hat{\lambda}^{(22)} & \dots & \hat{\lambda}^{(2K)} \\ \vdots & \vdots & \dots & \vdots \\ \hat{\lambda}^{(K1)} & \hat{\lambda}^{(K2)} & \dots & \hat{\lambda}^{(KK)} \end{bmatrix}$$

and let $\hat{P} = \hat{X} U I_{p,q} U^\top \hat{X}^\top$*, where* $\hat{X}$ *is defined from* $\hat{\Lambda}$ *and* U *is defined from* K *as in theorem 4.1. Equation* (11) *then implies*

$$\frac{1}{n}\|\rho_n^{-1}\hat{P} - \rho_n^{-1}P\|_F = O((n\rho_n)^{-1/2}), \quad \max_{ij} |\rho_n^{-1}\hat{P}_{ij} - \rho_n^{-1}P_{ij}| = O((n\rho_n)^{-1/2}) \tag{12}$$

with high probability.

Equation (11) guarantees that $n^{-1/2}\|\rho_n^{-1/2}\hat{\Lambda} - \Lambda\|_F = O((n\rho_n)^{-1/2})$. Equation (12) then guarantees that the mean square error for $\rho_n^{-1}(\hat{P} - P)$ converges to 0 almost surely and furthermore the entries of $\rho_n^{-1}\hat{P}$ converge uniformly to the entries of $\rho_n^{-1}P$; recall that $\rho_n^{-1}P_{ij} = \lambda_{iz_j}\lambda_{jz_i}$. We note that these results are stronger than existing results in Sengupta and Chen (2018); for example theorem 2 in Sengupta and Chen (2018) only guarantees $n^{-1/2}\|\rho_n^{-1/2}\hat{\Lambda} - \Lambda\|_F = o(1)$ as $n \to \infty$.

4.3 Simulation Study

4.4 Applications

5 Generalized Random Dot Product Graphs with Nonlinear Community Structure

5.1 Community Detection as Clustering in the Latent Space

As established in section 3, all Bernoulli random graphs are generalized random dot product graphs. Whether this is useful for community detection in block models depends entirely on the configuration of the latent vectors. In the case of the Erdös-Rényi model, SBM, DCBM, and PABM, each community forms a linear structure, i.e., a subspace, and this rigid structure can be exploited for community detection. In this section, we explore community-wise nonlinear latent structures.

5.2 The Manifold Block Model

Definition 5.1 (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 $\mathcal{M}_1, ..., \mathcal{M}_K \in \mathcal{X}$ for $\mathcal{X} = \{x, y \in \mathbb{R}^d : x^\top I_{p,q} y \in [0, 1]\}$ each by continuous function $g_k : [0, 1]^r \to \mathcal{X}$. Define probability distributions $F_1, ..., F_K$ each with support $[0, 1]^r$. Then the following mixture model is a *manifold block model*.

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

Example 5.1.

5.3 Algorithms for Nonintersecting Manifolds

5.4 Algorithms for Intersecting Manifolds

5.5 Examples and Simulations

5.6 Applications

5.7 Conclusions

Appendix A: Proofs of Theorems

Let V_n and $\hat{V}_n$ be the $n \times K^2$ matrices whose columns are the eigenvectors of P and A corresponding to the K^2 largest eigenvalues (in modulus), respectively. We first state an important technical lemma for bounding the maximum ℓ_2 norm difference between the rows of $\hat{V}_n$ and V_n. See Cape et al. (2019) and Rubin-Delanchy et al. (2017, Lemma 5) for a proof.

Lemma 5.1. *Let $A \sim \mathrm{PABM}(\{\lambda^{(k\ell\ell)}\}_K)$ be a K-blocks PABM graph on n vertices and let V and $\hat{V}$ be the $n \times K^2$ matrices whose columns are the eigenvectors of P and A corresponding to the K^2 largest eigenvalues in modulus, respectively. Let $v_i^\top$ and $\hat{v}_i^\top$ denote the ith row of V and $\hat{V}$, respectively. Then there exists a constant $c > 1$ and an orthogonal matrix W such that with high probability,*

$$\max_i \|W\hat{v}_i - v_i\| = O\Big(\frac{\log^c n}{n\sqrt{\rho_n}}\Big).$$

In particular we can take $c = 1 + \epsilon$ for any $\epsilon > 0$.

Proof of theorem 4.3. Recall the notations in lemma 5.1 and note that, under our assumption that the latent vectors $\lambda^{(k\ell)}$ are all homogeneous, we have $\max_i \|v_i\| = O(n^{-1/2})$.

Next recall theorem 4.2; in particular $B_{ij} = nv_i^\top v_j$. We therefore have

$$\begin{aligned}
\max_{ij} |\hat{B}_{ij} - B_{ij}| &= \max_{ij} n|\hat{v}_i^\top \hat{v}_j - v_i^\top v_j| \\
&\le n \max_{ij} |\hat{v}_i^\top W W^\top \hat{v}_j - v_i^\top v_j| \\
&\le n \max_{i,j} \Big(\|W^\top \hat{v}_i - v_i\| \times \|\hat{v}_j\| + \|W^\top \hat{v}_j - v_j\| \times \|v_i\| \Big) \\
&\le n \Big(\max_{ij} \|W\hat{v}_i - v_i\|^2 + \|W\hat{v}_i - v_i\| \times \|v_j\| + \|W\hat{v}_j - v_j\| \times \|v_i\| \Big) \\
&\le n \max_i \|W\hat{v}_i - v_i\|^2 + 2n \max_i \|W\hat{v}_i - v_i\| \times \max_j \|v_j\| \\
&= O\Big(\frac{\log^c n}{n^{1/2}\rho_n^{1/2}}\Big)
\end{aligned}$$

with high probability. Theorem 4.3 follows from the above bound together with the conclusion in theorem 4.2 that $B_{ij} = 0$ whenever vertices i and j belongs to different communities. □

We now provide a proof of theorem 4.4. Our proof is based on verifying the sufficient

conditions given in theorem 6 of Wang and Xu (2016) under which sparse subspace clustering based on solving the optimization problem in equation (9) yields an affinity matrix $B = |C| + |C^\top|$ satisfying the subspace detection property of definition 4.1. We first recall a few definitions used in Soltanolkotabi and Candés (2012) and Wang and Xu (2016); for ease of exposition, these definitions are stated using the notations of the current paper and we will drop the explicit dependency on n from our eigenvectors $\hat{V}$ of A and V of P.

Definition 5.2 (Inradius)**.** The inradius of a convex body $\mathcal{P}$, denoted by $r(\mathcal{P})$, is defined as the radius of the largest Euclidean ball inscribed in $\mathcal{P}$. Let X be a $n \times d$ matrix with rows $x_1, x_2, \ldots, x_n$. We then define, with a slight abuse of notation, $r(X)$ as the inradius of the convex hull formed by $\{\pm x_1, \pm x_2, \ldots, \pm x_n\}$.

Definition 5.3 (Subspace incoherence)**.** Let $\hat{V}$ be the eigenvectors of A corresponding to the K^2 largest eigenvalues in modulus. Let $\hat{V}^{(k)}$ denote the matrix formed by keeping only the rows of $\hat{V}$ corresponding to the k^{th} community and let $\hat{V}^{(-k)}$ denote the matrix formed by omitting the rows of $\hat{V}$ corresponding to the k^{th} community. Let $(\hat{v}_i^{(k)})^\top$ denote the ith row of $\hat{V}^{(k)}$ and $\hat{V}_{-i}^{(k)}$ be $\hat{V}^{(k)}$ with the i^{th} row omitted. Let V, $V^{(k)}$, $V^{(-k)}$, and $v_i^{(k)}$ be defined similarly using the eigenvectors V of P. Finally let $\mathcal{S}^{(k)}$ be the vector space spanned by the rows of $V^{(k)}$.

Now define $\nu_i^{(k)}$ for $k = 1, 2, \ldots, K$ and $i = 1, 2, \ldots, n_k$ as the solution of the following optimization problem

$$\nu_i^{(k)} = \max_{\eta} (\hat{v}_i^{(k)})^\top \eta - \frac{1}{2\lambda} \eta^\top \eta, \quad \text{subject to } \|V_{-i}^{(k)} \eta\|_\infty \leq 1.$$

Given $\nu_i^{(k)}$, let $\mathbb{P}_{\mathcal{S}^{(k)}}(\nu_i^{(k)})$ be the vector in $\mathbb{R}^{K^2}$ corresponding to the orthogonal projection of $\nu_i^{(k)}$ onto the vector space $\mathcal{S}^{(k)}$ and define the projected dual direction $w_i^{(k)}$ as

$$w_i^{(k)} = \frac{\mathbb{P}_{\mathcal{S}^{(k)}}(\nu_i^{(k)})}{\|\mathbb{P}_{\mathcal{S}^{(k)}}(\nu_i^{(k)})\|}.$$

Now let $W^{(k)} = [w_1^{(k)} \mid \cdots \mid w_{n_k}^{(k)}]^\top$ and define the subspace incoherence for $\hat{V}^{(k)}$ by

$$\mu^{(k)} = \mu(\hat{V}^{(k)}) = \max_{v \in V^{(-k)}} \|W^{(k)} v\|_\infty.$$

Proof of theorem 4.4. For a given $k = 1, 2 \ldots, K$, let $r^{(k)} = \min_i r(V_{-i}^{(k)})$ be inradius of the convex hull formed by the rows of $V_{-i}^{(k)}$ and let $r_* = \min_k r^{(k)}$. Then Theorem 6 in Wang and Xu (2016) states that there exists a $\lambda > 0$ such that $\sqrt{n}\hat{V}$ satisfies the subspace detection property in definition 4.1 whenever the following two conditions are satisfied:

$$\mu^{(k)} < r^{(k)} \quad \text{for all } k = 1, 2, \ldots, K, \tag{13}$$

$$\max_i \|W\hat{v}_i - v_i\| \leq \min_k \frac{r_*(r^{(k)} - \mu^{(k)})}{2 + 7r^{(k)}}. \tag{14}$$

We now verify that for sufficiently large n, equation (13) and Eq. (14) holds with high probability.

Verifying equation (13). If n is sufficiently large then there are enough vertices in each community k so that $\text{span}(V_{-i}^{(k)}) = \mathcal{S}^{(k)}$ for all i and hence $r^{(k)} = \min_i r(V_{-i}^{(k)}) > 0$ for all $k = 1, 2, \ldots, K$.

Next, by theorem 4.2 we have that the subspaces $\{\mathcal{S}^{(1)}, \ldots, \mathcal{S}^{(K)}\}$ are mutually orthogonal, i.e., $v^\top w = 0$ for all $v \in \mathcal{S}^{(k)}$ and $w \in \mathcal{S}^{(\ell)}$ with $k \neq \ell$. Now let $z \in \mathbb{R}^{K^2}$ be arbitrary and let $\tilde{z} = \mathbb{P}_{\mathcal{S}^{(k)}} z$ be the projection of z onto $\mathcal{S}^{(k)}$. We then have $v^\top \tilde{z} = 0$ for all $v \in V^{(-k)}$. Because z is arbitrary, this implies $\|W^{(k)} v\|_\infty = 0$ for all $v \in V^{(-k)}$ and hence $\mu^{(k)} = 0$ for all $k = 1, 2, \ldots, K$. Therefore $\mu^{(k)} < r^{(k)}$ for all $k = 1 = 2, \ldots, K$ as desired.

Verifying equation (14). Let $\delta = \max_i \sqrt{n}\|W\hat{v}_i - v_i\|$. Then from lemma 5.1, we have $\delta \xrightarrow{a.s.} 0$ and hence

$$\delta < \min_k \frac{r_*(r^{(k)} - \mu^{(k)})}{2 + 7r^{(k)}}$$

asymptotically almost surely.

In summary $\sqrt{n}\hat{V}$ satisfies the subspace detection property with probability converging to 1 as $n \to \infty$. □

Remark. Theorem 6 of Wang and Xu (2016) assumes that each row v_i of V has unit norm, i.e., $\|v_i\| = 1$ for all i. This assumption has the effect of scaling the $r^{(k)}$ so that $r^{(k)} \leq 1$ for all $k = 1, 2, \ldots, K$. We emphasize that this assumption has no effect on the proof of Theorem 4.4. Indeed, because $\mu^{(k)} = 0$ for all k, as long as the rows of $V^{(k)}$ spans the subspace $\mathcal{S}^{(k)}$, then

$ar^{(k)} > \mu^{(k)}$ for any scalar $a > 0$.

Proof of Theorem 4.5. Let P be organized by community such that $P^{(k\ell)}$ denote the $n_k \times n_\ell$ matrix obtained by keeping only the rows of P corresponding to vertices in community k and the columns of P corresponding to vertices in community ℓ. We define $A^{(k\ell)}$ analogously. Recall that $P^{(k\ell)} = \lambda^{(k\ell)}(\lambda^{(\ell k)})^\top$ for all k, ℓ. We now consider estimation of $P^{(k\ell)}$ for the cases when $k = \ell$ versus when $k \neq \ell$.

Case $k = l$. Let $P^{(kk)} = \sigma_{kk}^2 u^{(kk)}(u^{(kk)})^\top$ be the singular value decomposition of $P^{(kk)}$. We can then define $\tilde{\lambda}^{(kk)} = \sigma_{kk} u^{(kk)}$. Now let $\hat{U}^{(kk)}\hat{\Sigma}^{(kk)}(\hat{U}^{(kk)})^\top$ be the singular value decomposition of $A^{(kk)}$, and let $\hat{\sigma}_{kk}^2 \hat{u}^{(kk)}(\hat{u}^{(kk)})^\top$ be the best rank-one approximation of $A^{(kk)}$. Define $\hat{\lambda}^{(kk)} = \hat{\sigma}_{kk}\hat{u}^{(kk)}$. Then $\hat{\lambda}^{(kk)}$ is the adjacency spectral embedding approximation of $\lambda^{(kk)}$, and by Theorem 5 of Rubin-Delanchy et al. (2017), we have

$$\|\hat{\lambda}^{(kk)} - \lambda^{(kk)}\|_\infty = O\Big(\frac{\log n_k}{\sqrt{n_k}}\Big)$$

with high probability. Here $\|\cdot\|_\infty$ denote the ℓ_∞ norm of a vector.

Case $k \neq l$. Let $P^{(k\ell)} = \sigma_{k\ell}^2 u^{(k\ell)}(v^{(k\ell)})^\top$ and $P^{(\ell k)} = \sigma_{kl}^2 u^{(\ell k)}(v^{(\ell k)})^\top$ be the singular value decompositions and note that $\sigma_{k\ell} = \sigma_{\ell k}$, $u^{(k\ell)} = v^{(\ell k)}$, and $v^{(k\ell)} = u^{(\ell k)}$. Now define $\lambda^{(k\ell)} = \sigma_{k\ell} u^{(k\ell)}$ and $\lambda^{(\ell k)} = \sigma_{k\ell} v^{(k\ell)}$.

Next consider the Hermitian dilation

$$M^{(k\ell)} = 2\begin{bmatrix} 0 & P^{(k\ell)} \\ P^{(\ell k)} & 0 \end{bmatrix}$$

which is a symmetric $(n_k + n_\ell) \times (n_k + n_\ell)$ matrix. The eigendecomposition of $M^{(k\ell)}$ is then

$$M^{(k\ell)} = \begin{bmatrix} u^{(k\ell)} & -u^{(k\ell)} \\ v^{(k\ell)} & v^{(k\ell)} \end{bmatrix} \times \begin{bmatrix} \sigma_{kl}^2 & 0 \\ 0 & -\sigma_{kl}^2 \end{bmatrix} \times \begin{bmatrix} u^{(k\ell)} & -u^{(k\ell)} \\ v^{(k\ell)} & v^{(k\ell)} \end{bmatrix}^\top$$

Thus treating $M^{(k\ell)}$ as the edge probability matrix of a GRDPG, we have latent positions in $\mathbb{R}^2$

given by the $(n_k + n_\ell) \times 2$ matrix

$$\Lambda^{(k\ell)} = \begin{bmatrix} \sigma_{k\ell} u^{(k\ell)} & \sigma_{k\ell} u^{(k\ell)} \\ \sigma_{k\ell} v^{(k\ell)} & -\sigma_{k\ell} v^{(k\ell)} \end{bmatrix} = \begin{bmatrix} \lambda^{(k\ell)} & \lambda^{(k\ell)} \\ \lambda^{(\ell k)} & -\lambda^{(\ell k)} \end{bmatrix}.$$

Now consider

$$\hat{M}^{(k\ell)} = \begin{bmatrix} 0 & A^{(k\ell)} \\ A^{(\ell k)} & 0 \end{bmatrix}$$

We can then view $\hat{M}^{(k\ell)}$ as an adjacency matrix drawn from the edge probabilities matrix $M^{(k\ell)}$. Now suppose that the adjacency spectral embedding of $\hat{M}^{(k\ell)}$ is represented as the $(n_k + n_\ell) \times 2$ matrix

$$\hat{\Lambda}^{(k\ell)} = \begin{bmatrix} \hat{\lambda}^{(k\ell)} & \hat{\lambda}^{(k\ell)} \\ \hat{\lambda}^{(\ell k)} & -\hat{\lambda}^{(\ell k)} \end{bmatrix}$$

where each $\hat{\lambda}^{(k\ell)}$ is defined as in Algorithm 3. Then by Theorem 5 of Rubin-Delanchy et al. (2017), there exists an indefinite orthogonal transformation W^* such that, with high probability,

$$\max_i |W^* \hat{\Lambda}_i^{(k\ell)} - \Lambda_i^{(k\ell)}\| = O\Big(\frac{\log(n_k + n_\ell)}{\sqrt{n_k + n_\ell}}\Big)$$

with high probability. Here $\Lambda_i^{(k\ell)}$ and $\hat{\Lambda}_i^{(k\ell)}$ denote the ith rows of $\Lambda^{(k\ell)}$ and $\hat{\Lambda}^{(k\ell)}$, respectively.

Furthermore, by looking at the proof of Theorem 5 in (Rubin-Delanchy et al., 2017), we see that W^* is also blocks diagonal with 2 blocks where the positive eigenvalues of $M^{(k\ell)}$ forming a block and the negative eigenvalues of $M^{(k\ell)}$ forming the remaining block. Because $M^{(k\ell)}$ has one positive eigenvalue and one negative eigenvalue, we see that W^* is necessarily of the form $W^* = \left[\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right]$ Using this form for W^*, we obtain

$$\max\{\|\hat{\lambda}^{(k\ell)} - \lambda^{(k\ell)}\|_\infty, \|\hat{\lambda}^{(\ell k)} - \lambda^{(\ell k)}\|_\infty\} = O\Big(\frac{\log(n_k + n_\ell)}{\sqrt{n_k + n_\ell}}\Big)$$

with high probability. Combining this bound with the bound for $\|\hat{\lambda}^{(kk)} - \lambda^{(kk)}\|_\infty$ given above yields equation (11) in theorem 4.5. □

References

Abbe, E. (2018). Community detection and stochastic block models: Recent developments. *Journal of Machine Learning Research*, 18(177):1–86.

Airoldi, E. M., Blei, D. M., Fienberg, S. E., and Xing, E. P. (2009). Mixed membership stochastic blockmodels. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 21*, pages 33–40. Curran Associates, Inc.

Cape, J., Tang, M., and Priebe, C. E. (2019). Signal-plus-noise matrix models: eigenvector deviations and fluctuations. *Biometrika*, 106:243–250.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

Elhamifar, E. and Vidal, R. (2009). Sparse subspace clustering. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2790–2797.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458):611–631.

Gilbert, E. N. (1959). Random Graphs. *The Annals of Mathematical Statistics*, 30:1141 – 1144.

Jones, J. M. (2022). LGBT identification in U.S. ticks up to 7.1%. *Gallup*.

Karrer, B. and Newman, M. E. J. (2011). Stochastic blockmodels and community structure in networks. *Physical Review E*, 83(1).

Kolaczyk, E. and Csárdi, G. (2014). *Statistical Analysis of Network Data with R*. Use R! Springer New York.

Le, C. M., Levina, E., and Vershynin, R. (2016). Optimization via low-rank approximation for community detection in networks. *Ann. Statist.*, 44(1):373–400.

Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., and Ma, Y. (2013). Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:171–184.

Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137.

Lorrain, F. and White, H. C. (1971). Structural equivalence of individuals in social networks. *The Journal of Mathematical Sociology*, 1(1):49–80.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In Cam, L. M. L. and Neyman, J., editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press.

Nasihatkon, B. and Hartley, R. (2011). Graph connectivity in sparse subspace clustering. In *Computer Vision and Pattern Recognition*, pages 2137–2144.

Nepusz, T., Petróczi, A., Négyessy, L., and Bazsó, F. (2008). Fuzzy communities and the concept of bridgeness in complex networks. *Phys. Rev. E*, 77:016107.

Noroozi, M. and Pensky, M. (2022). The hierarchy of block models. *Sankhya A*, 84(1):64–107.

Noroozi, M., Rimal, R., and Pensky, M. (2019). Estimation and clustering in popularity adjusted block model. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, n/a(n/a).

Rubin-Delanchy, P., Cape, J., Tang, M., and Priebe, C. E. (2017). A statistical interpretation of spectral embedding: the generalised random dot product graph.

Sengupta, S. and Chen, Y. (2018). A block model for node popularity in networks with community structure. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 80(2):365–386.

Soltanolkotabi, M. and Candés, E. J. (2012). A geometric analysis of subspace clustering with outliers. *Ann. Statist.*, 40(4):2195–2238.

Sussman, D. L., Tang, M., Fishkind, D. E., and Priebe, C. E. (2012). A consistent adjacency spectral embedding for stochastic blockmodel graphs. *Journal of the American Statistical Association*, 107:1119–1128.

von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416.

Wang, Y.-X. and Xu, H. (2016). Noisy sparse subspace clustering. *Journal of Machine Learning Research*, 17(12):1–41.

Young, S. J. and Scheinerman, E. R. (2007). Random dot product graph models for social networks. In Bonato, A. and Chung, F. R. K., editors, *Algorithms and Models for the Web-Graph*, pages 138–149, Berlin, Heidelberg. Springer Berlin Heidelberg.