Comodularity and detection of co-communities

Thomas E. Bartlett*
Department of Statistical Science, University College London, London WC1E 7HB, United Kingdom

(Received 1 February 2021; accepted 8 November 2021; published 29 November 2021)

This paper introduces the notion of comodularity, to cocluster observations of bipartite networks into co-communities. The task of coclustering is to group together nodes of one type with nodes of another type, according to the interactions that are the most similar. The measure of comodularity is introduced to assess the strength of co-communities, as well as to arrange the representation of nodes and clusters for visualization, and to define an objective function for optimization. We demonstrate the usefulness of our proposed methodology on simulated data, and with examples from genomics and consumer-product reviews.

DOI: 10.1103/PhysRevE.104.054309

I. INTRODUCTION

Networks are used to parsimoniously represent relationships between entities of the same type. Classical analysis methods use parametric models of network data, such as degree-based and/or community-based models [1–6]. The last few years have seen a flurry of activity in statistical network analysis (see, for example, [7–10]). One of the best-studied tools is the stochastic blockmodel [1,2], and various extensions to it [11–13]; various methods of fitting this model have been proposed, where maximizing modularity remains an important practical approach [2,14]. Recent work in clustering network nodes has generalized the applicability of the stochastic blockmodel by showing that arbitrary exchangeable networks can be represented using a blockmodel [2,15,16]; such a representation is called a "network histogram." The network histogram method [16] can be used to estimate the optimal granularity at which communities, or functional subnetwork modules, can be approximated and isolated in social and biological networks, i.e., to estimate the optimal number of clusters or communities of network nodes. Alternatively, several Bayesian approaches to estimating the optimal number of communities in a network have also been proposed [17–19]. However, it is well established that when clustering is implemented, estimating the optimal number of clusters is an important and separate problem from the design of the clustering methodology [20]. For example, sophisticated solutions to this problem such as the gap statistic [21] propose methodology for estimating the optimal number of clusters, and this is done independently from the choice of clustering methodology. In this paper we focus on the problem of clustering methodology for variables of different types, i.e., coclustering.

Studying relationships between variables of the same type is naturally very useful; its simplest generalization is to study relationships between variables of a different type; this is known as the coclustering problem [22–25], and is of much current interest in application areas from genomics to natural language processing [26–28]. The coclustering problem can also be approached nonparametrically, as is made clear by [23] and [25]. We start from the modularity approach to recognizing communities [14], realizing that extending such understanding to variables of different types is nontrivial [24,29]. Having recognized communities in both types of variables, we need to transform the clustering or grouping of both types of variables into an ordering of groups. This is not inherent to the formulation of the Aldous-Hoover representation of the generating mechanism of the random array we are modeling, but is important for visualization purposes. We aggregate the modularity over the groupings of each of the two types of variables (corresponding to *rows* or *columns*) to guide this choice of visualization. We also use modularity to compare co-communities, which we define as pairings of one group or cluster of each type of node, providing a unique paradigm for understanding these important bipartite network structures. Finally, to demonstrate the usefulness of our proposed methodology, we analyze two characteristic network data sets, as follows: a genomics data set, in which a co-community represents a functional module that involves two types of genomic features (i.e., measured on two different data modalities or platforms), and a movie review data set, in which a co-community represents a set of movies (probably with similar characteristics) enjoyed by a particular group of (possibly similar) people. Importantly, in our model setup, it is possible for a node to be part of one co-community, or part of multiple co-communities, or part of no co-communities. Thus, we show how our proposed analysis methods enable us to discover both known and hitherto unknown characteristics of these two data sets. This paper is organized as follows: Section II defines the stochastic blockmodel, and gives the representation of an arbitrary separately exchangeable array. It also defines the comodularity, and explains how the array data will be analyzed. Then Sec. III shows how to find the co-communities in data, and Sec. IV gives examples to illustrate the performance of our proposed methodology. The Appendixes provide all proofs of the paper.

*thomas.bartlett.10@ucl.ac.uk

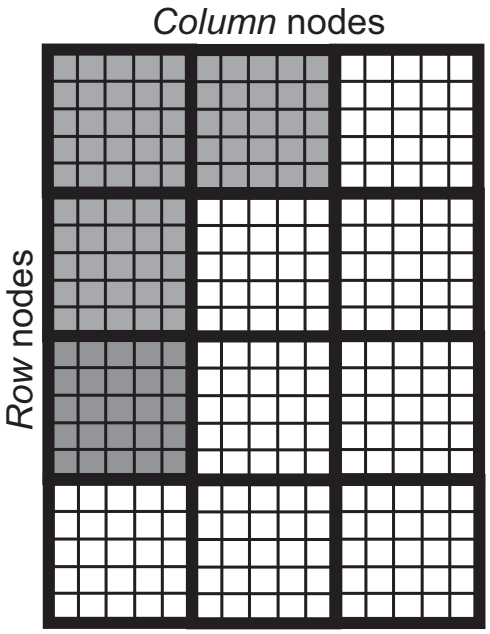

FIG. 1. Co-community structure. The margins of the random array represent different types of nodes, and the array elements define the edges of the bipartite network. The shaded blocks represent co-communities of the two different types of nodes

II. COMODULARITY AND CO-COMMUNITY DETECTION

We begin this section by defining the degree-corrected stochastic coblockmodel [22,23,30] together with notation. We then give a definition of the Newman-Girvan modularity [31] and, by analogy, we define a quantity which we term the "comodularity," and we specify an algorithm for maximizing this quantity. While previous work (see, e.g., that by [32]) has separately identified groupings of the different types of nodes, the notion of "comodularity" considers the pairing into blocks of these groupings of different types of nodes (an issue which does not arise in networks with only one type of node). We call such a pairing a "co-community," illustrated by the shaded blocks in Fig. 1. We show that under certain conditions, maximizing the comodularity in this way is equivalent to maximizing the model likelihood of the specified degree-corrected stochastic coblockmodel.

Definition 1 (Degree-corrected stochastic coblockmodel). For $m, l \in \mathbb{N}^+$, assign the labeling for the set of X nodes as $\{1, \ldots, m\}$, and for the set of Y nodes $\{1, \ldots, l\}$, where this labeling is chosen without loss of generality. Denote an X-node grouping as $g_p^{(X)} \in G^{(X)}$, $p \in \{1, \ldots, k^{(X)}\}$, and a Y-node grouping as $g_q^{(Y)} \in G^{(Y)}$, $q \in \{1, \ldots, k^{(Y)}\}$, where $G^{(X)}$ and $G^{(Y)}$ are exhaustive lists of mutually exclusive X- and Y-node groupings, respectively. Define map functions $z^{(X)}(i)$ and $z^{(Y)}(j)$, such that $g_p^{(X)} = \{i : z^{(X)}(i) = p\}$ and $g_q^{(Y)} = \{j : z^{(Y)}(j) = q\}$. Define co-community connectivity parameters $\boldsymbol{\theta} \in [0,1]^{k^{(X)} \times k^{(Y)}}$, where $\theta_{z^{(X)}(i), z^{(Y)}(j)}$ is the propensity of X-node i in group $z^{(X)}(i)$ to form a connection with Y-node j in group $z^{(Y)}(j)$. Define also node-specific connectivity parameters $\boldsymbol{\pi}^{(X)} \in \mathbb{R}^m_{\geqslant 0}$ and $\boldsymbol{\pi}^{(Y)} \in \mathbb{R}^l_{\geqslant 0}$. Let the elements of the adjacency matrix $\mathbf{A} \in \{0,1\}^{m \times l}$ follow the law of

$$A_{ij} \sim \text{Bernouilli}\left(\pi_i^{(X)} \pi_j^{(Y)} \theta_{z^{(X)}(i), z^{(Y)}(j)}\right), \quad 1 \leqslant i \leqslant m, \ 1 \leqslant j \leqslant l. \tag{1}$$

Then, we call the generative mechanism of A_{ij} the "degree corrected stochastic coblockmodel."

We note that the terminology "X nodes" and "Y nodes" is nonstandard; we introduce it here to increase clarity of exposition. To improve identifiability of parameters of the model in Definition 1, and defining a co-community as a pairing of the X-node grouping $g_p^{(X)}$ with the Y-node grouping $g_q^{(Y)}$, we introduce a special case of the blockmodel favored by many other authors [33], that $\theta_{z^{(X)}(i), z^{(X)}(j)}$ may take only two values:

$$\theta_{p,q} = \begin{cases} \theta_{\text{in}} & \text{if the pairing of } X\text{-node grouping } g_p^{(X)} \text{ with} \\ & Y\text{-node grouping } g_q^{(Y)} \text{ is a co-community,} \\ \theta_{\text{out}} & \text{otherwise.} \end{cases} \tag{2}$$

We can also replace the Bernoulli model likelihood with a Poisson likelihood: because the Bernoulli success probability is typically small, and the number of potential edges (i.e., pairings of nodes) is large, a Poisson distribution with the same mean converges to the same distribution, and so it makes little difference in practice [11,34]. Its usage greatly simplifies the technical derivations. Hence, we calculate the model log likelihood as follows (assuming $A_{ij} \in \{0,1\}$ and therefore $A_{ij}! = 1$ for all i, j):

$$\begin{aligned} &\ell(\boldsymbol{\theta}, \boldsymbol{\pi}^{(X)}, \boldsymbol{\pi}^{(Y)}; G^{(X)}, G^{(Y)}) \\ &= \sum_{i=1}^{m} \sum_{j=1}^{l} A_{ij} \ln\left(\pi_i^{(X)} \pi_j^{(Y)} \theta_{z^{(X)}(i), z^{(Y)}(j)}\right) \\ &\quad - \pi_i^{(X)} \pi_j^{(Y)} \theta_{z^{(X)}(i), z^{(Y)}(j)}. \end{aligned} \tag{3}$$

The Newman-Girvan modularity [31] measures, for a particular partition of a network into communities, the observed number of edges between community members, compared to the expected number of edges between community members without the community partition with the degree correction. The Newman-Girvan modularity may be defined as follows:

Definition 2 (Newman-Girvan modularity). Define $\mathbf{A} \in \{0,1\}^{n \times n}$ as a symmetric adjacency matrix representing a unipartite network with nodes $i \in \{1, \ldots, n\}$, define $\mathbf{d}$ as the degree vector of the nodes of this network, $d_i = \sum_{j=1}^n A_{ij}$, and define the normalizing factor d^{++} as twice the total number of edges, $d^{++} = \sum_{i=1}^n d_i$. Define a community, or grouping, of nodes as $g \in G$, where G represents the set of all such groupings of nodes, define the map function $z(i)$ such that $g_a = \{i : z(i) = a\}$, and let $\mathbb{I}[z(i) = z(j)]$ specify whether nodes i and j appear together in any community g, such that

$$\mathbb{I}[z(i) = z(j)] = \begin{cases} 1 & \text{if nodes } i \text{ and } j \text{ are grouped together} \\ & \text{in any community } g \in G \\ 0 & \text{otherwise.} \end{cases}$$

Then, the Newman-Girvan (NG) modularity Q_{NG} is defined as

$$Q_{\text{NG}} = \frac{1}{d^{++}} \sum_{i=1}^{n} \sum_{j=1}^{n} \left[A_{ij} - \frac{d_i d_j}{d^{++}}\right] \mathbb{I}[z(i) = z(j)]. \tag{4}$$

The comodularity is then defined by analogy with the Newman-Girvan modularity (Definition 2) as follows:

Definition 3 (Comodularity). With m and l given by Definition 1 and $\mathbf{A}$ generated according to Definition 1, define $\mathbf{d}^{(X)}$ and $\mathbf{d}^{(Y)}$ as the degree vectors of the X and Y nodes of the network, $d_i^{(X)} = \sum_{j=1}^{l} A_{ij}$ and $d_j^{(Y)} = \sum_{i=1}^{m} A_{ij}$, and define the normalizing factor d^{++} as twice the total number of edges, $d^{++} = \sum_{i=1}^{m} d_i^{(X)} = \sum_{j=1}^{l} d_j^{(Y)}$. With $g^{(X)}$ and $g^{(Y)}$, $z^{(X)}$ and $z^{(Y)}$ also defined in direct analog according to Definition 1, let $c_t = \{p, q\} \in C, t = \{1, \ldots, T\}$. The enumeration of the pair $\{p, q\}$ is arbitrary, and is used to facilitate ease of access of the coblocks in a chosen order. The coblock c_t specifies that the X-node grouping $g_p^{(X)}$ is paired with the Y-node grouping $g_q^{(Y)}$; we refer to such a pairing as a "co-community." Furthermore, let $\Psi(C; G^{(X)}, G^{(Y)}; i, j) \in \{0, 1\}$ specify whether nodes i and j appear together in any co-community $c \in C$, such that

$$\Psi(C; G^{(X)}, G^{(Y)}; i, j) = \begin{cases} 1 & \text{if } \{z^{(X)}(i), z^{(Y)}(j)\} = c : c \in C \\ 0 & \text{otherwise.} \end{cases}$$

Then, the comodularity Q_{XY} is defined as

$$Q_{XY} = \frac{1}{d^{++}} \sum_{i=1}^{m} \sum_{j=1}^{l} \left[A_{ij} - \frac{d_i^{(X)} d_j^{(Y)}}{d^{++}} \right] \Psi(C; G^{(X)}, G^{(Y)}; i, j). \tag{5}$$

We note that for the comodularity (unlike the Newman-Girvan modularity) we require a set of pairings of X-node groupings with Y-node groupings C, such that each $c_t \in C$ is a pairing of an X-node grouping $g_p^{(X)} \in G^{(X)}$ with a Y-node grouping $g_q^{(Y)} \in G^{(Y)}$. Also, due to the asymmetry of the coclustering problem, $c_t = \{p, q\} \neq \{q, p\}$. This separately specified set of parings C is not required in the case of the Newman-Girvan modularity, because in the unipartite network setting, there is only one type of node, and hence node groupings already "match up" with one another. This can be visualized, in the unipartite network setting, as community structure present along the leading diagonal of the adjacency matrix, if the nodes are ordered by community. In the co-community setting, an X-node grouping $g^{(X)}$ may be paired in C with many, with one, or with no Y-node groupings $g^{(Y)} \in G^{(Y)}$, and equivalently a Y-node grouping $g^{(Y)}$ may be paired in C with many, with one, or with no X-node groupings $g^{(X)} \in G^{(X)}$ (Fig. 1). Further, if the X nodes and Y nodes of the network are arranged in the adjacency matrix according to the groupings $g^{(X)}$ and $g^{(Y)}$, there is no reason co-communities should appear along the leading diagonal. Hence, the function Ψ in Eq. (5) generalizes the role of the indicator function in Eq. (4). We note that other approaches to this problem directly specify a null model [35]. We also note that sometimes in practice we must relax the requirement of $\Psi \in \{0, 1\}$; the reason for this is made clear in the technical derivations (for tractability) in Appendix A which relate to Algorithm 1 (which follows next).

Community detection of k communities can be performed by fitting the degree-corrected stochastic blockmodel. This is equivalent, under many circumstances, to spectral clustering [2,33,36], which may be carried out by grouping the nodes into k clusters in the space of the eigenvectors corresponding to the second to kth greatest eigenvalues of the Laplacian $\mathbf{L} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$, where $\mathbf{D}$ is the diagonal matrix of the degree distribution. Co-community detection in a bipartite network of nodes attributed to the variables X and Y (respectively, X nodes and Y nodes) can equivalently be performed by degree-corrected spectral clustering [32].

A procedure to find an assignment of X and Y nodes to $k^{(X)}$ X-node groupings ("row clusters") and $k^{(Y)}$ Y-node groupings ("column clusters"), respectively, which finds a (possibly locally) optimum value of the comodularity Q_{XY}, is specified in Algorithm 1:

Algorithm 1. With $\mathbf{A}$ and Q_{XY} defined as in Definition 1, and $\mathbf{d}^{(X)}$ and $\mathbf{d}^{(Y)}$ defined as in Definition 3:

(1) Calculate the co-Laplacian $\mathbf{L}_{XY}$ [32] as

$$\mathbf{L}_{XY} = (\mathbf{D}^{(X)})^{-1/2}\mathbf{A}(\mathbf{D}^{(Y)})^{-1/2}, \tag{6}$$

where $\mathbf{D}^{(X)}$ and $\mathbf{D}^{(Y)}$ are the diagonal matrices of $\mathbf{d}^{(X)}$ and $\mathbf{d}^{(Y)}$, respectively.

(2) Calculate the singular value decomposition (SVD) of the co-Laplacian $\mathbf{L}_{XY}$.

(3) Separately cluster the X and Y nodes in the spaces of the left and right singular vectors corresponding to the second to $k^{(X)}$th andsecond to $k^{(Y)}$th greatest singular values, respectively, of this SVD of $\mathbf{L}_{XY}$.

(4) Identify the set of co-communities C, as pairings of particular X-node groupings and Y-node groupings.

We note that there are a range of choices and alternatives for steps 1–3 of Algorithm 1. The specifics of how to identify the set of co-communities in step 4 of Algorithm 1, i.e., the identification of C as particular pairings of the identified X-node groupings and Y-node groupings, are discussed further in Sec. III. Technical derivations relating to Algorithm 1 appear in Appendix A, and are based on arguments made previously in the context of unipartite (symmetric) community detection [33] for two communities, extending them to this context of (asymmetric) co-community detection. We note in particular that the notion of modularity assumes that within-community edges are more probable than between-community edges, and therefore modularity maximization is only consistent if constraints are applied to ensure this assumption holds [11]. In the community detection setting, under suitable constraints, the solutions which maximize model likelihood and modularity are identical [2].

Proposition 1. The solution which maximizes the model likelihood specified in Eq. (3), subject also to the constraint of Eq. (2), is equivalent to the maximum comodularity assignment obtained via Algorithm 1.

Proof. The proof for the case of two co-communities appears in Appendix B. It extends arguments made previously in relation to community detection [33] to this context of co-community detection. ■

III. IDENTIFICATION AND COMPARISON OF CO-COMMUNITIES

Fitting the stochastic coblockmodel by spectral clustering as described in Algorithm 1 involves using k means to cluster the X and Y nodes in the spaces of the left and right singular vectors of the co-Laplacian [Eq. (6)]. However, there is subsequently the problem of how to identify the co-communities,

as represented by the shaded blocks in Fig. 1. This problem of identifiability is novel: it does not arise when fitting the stochastic blockmodel to unipartite networks by spectral clustering, because of the symmetry of the problem (if there is only one type of node, then different types of nodes do not need to be grouped). Finding the co-communities (illustrated by the shaded blocks in Fig. 1) consists of estimating the set C (Definition 3) of pairings of X-node groupings $g^{(X)} \in G^{(X)}$ with Y-node groupings $g^{(Y)} \in G^{(Y)}$. It replaces identifying the "diagonal" of the blockmodel, a concept that is less straightforward than for a symmetric adjacency matrix. In this section we propose a methodology to address this problem in a fully automated way in the bipartite network setting, along with related issues of visualization and optimization.

Fitting the symmetric blockmodel in the unipartite community detection setting, there are exactly $k = k^{(X)} = k^{(Y)}$ communities (because of symmetry). Each row grouping matches up with exactly one column grouping, because the row and column groupings are the same thing. On the other hand, fitting the asymmetric coblockmodel by spectral clustering as in Algorithm 1 leads to $k^{(X)}$ and $k^{(Y)}$ row and column clusters. Hence, these $k^{(X)}$ and $k^{(Y)}$ row and column clusters provide $k^{(X)} \times k^{(Y)}$ potential co-communities. Which of these are significant (as in the shaded blocks in Fig. 1)? The best-known solution to this problem clusters all the nodes at once (after concatenating the left and right singular vectors) [32], instead of clustering the X and Y nodes separately. However, that approach requires $k^{(X)} = k^{(Y)}$, so that each X-node grouping is paired with exactly one Y-node grouping. Our approach does not have this restriction, and hence can model a broader class of bipartite network structures, by introducing the notion of co-communities, as illustrated in Fig. 1. On the other hand, our approach must answer the question, how should we assess and compare the $k^{(X)} \times k^{(Y)}$ potential co-communities? That is, how should we compare each different pairing of an estimated X-node grouping $\hat{g}^{(X)} \in \hat{G}^{(X)}$, with an estimated Y-node grouping $\hat{g}^{(Y)} \in \hat{G}^{(Y)}$, to provide an assignment of the X nodes and Y nodes to co-communities? In practice, we expect the number of co-communities, $T = |C|$ (where $|\cdot|$ represents cardinality), to be significantly less than $k^{(X)} \times k^{(Y)}$. In the unipartite community detection setting, $k^{(X)} = k^{(Y)} = k$, and only the blocks on the diagonal can be communities: hence in effect there we have $T = k = \sqrt{k^{(X)} \times k^{(Y)}}$.

To estimate the set of co-communities, $c_t \in C$, $t = \{1, \ldots, T\}$, in this bipartite network setting, we calculate the "local comodularity" for each pairing $\hat{g}^{(X)}$ with $\hat{g}^{(Y)}$, by considering a relevant subpart of the comodularity matrix $\mathbf{B}$ [Eq. (7)]:

Definition 4 (Local comodularity). With $\mathbf{A}$ given by Definition 1, with $\mathbf{d}^{(X)}$, $\mathbf{d}^{(Y)}$ and d^{++} given by Definition 3, with

$$B_{ij} = A_{ij} - \frac{d_i^{(X)} d_j^{(Y)}}{d^{++}}, \quad \mathbf{B} = \mathbf{A} - \frac{1}{d^{++}} \mathbf{d}^{(X)} (\mathbf{d}^{(Y)})^{\top}, \tag{7}$$

and with the set of X-node groupings and the set of Y-node groupings estimated according to Algorithm 1 as $\hat{G}^{(X)}$ and $\hat{G}^{(Y)}$, respectively, where $|\hat{G}^{(X)}| = k^{(X)}$ and $|\hat{G}^{(Y)}| = k^{(Y)}$, where $|\cdot|$ represents cardinality, for a particular pairing of estimated X-node grouping $\hat{g}^{(X)} \in \hat{G}^{(X)}$ with estimated Y-node grouping $\hat{g}^{(Y)} \in \hat{G}^{(Y)}$, the local comodularity $Q_{XY}(\hat{g}^{(X)}, \hat{g}^{(Y)})$ is defined as

$$Q_{XY}(\hat{g}^{(X)}, \hat{g}^{(Y)}) = \frac{1}{d^{++}} \sum_{i \in \hat{g}^{(X)}} \sum_{j \in \hat{g}^{(Y)}} B_{ij}. \tag{8}$$

"Local" here means that we are considering a statistic for an individual block, out of the many blocks which are found in general along each row and column. Each of the $k^{(X)} \times k^{(Y)}$ possible pairings of $\hat{g}^{(X)}$ with $\hat{g}^{(Y)}$ can be defined, or not, as a co-community; doing so means that they are included in, or excluded from, the estimated set of co-communities $\hat{C}$ (Definition 3). To consider all permutations, $2^{k^{(X)} \times k^{(Y)}}$ such assignments would need to be considered, which would be computationally very demanding. However, this problem can be avoided by defining summary statistics targeted for particular purposes. The three such purposes which we consider here are described in the following subsections: Sec. III A, comparing potential co-communities and assessing their strength; Sec. III B, arranging the co-communities for visualization; and Sec. III C, defining an algorithmic objective function to be optimized, when determining co-community partitions.

A. Comparing and assessing significance of co-communities

Under a null model of no co-community structure, $\theta_{z^{(X)}(i), z^{(Y)}(j)} = \text{const}$, for all i, j. Therefore, referring to the log-linear model [34], Eq. (1) becomes

$$A_{ij} \sim \text{Bernouilli}\left(\frac{\pi_i^{(X)} \pi_j^{(Y)}}{\pi^{++}} \right), \tag{9}$$

where we have defined

$$\theta_{z^{(X)}(i), z^{(Y)}(j)} = 1/\pi^{++}. \tag{10}$$

Hence under this null,

$$\mathbb{E}(A_{ij}) = \frac{\pi_i^{(X)} \pi_j^{(Y)}}{\pi^{++}},$$

which implies that for large networks which are not too sparse, $\mathbb{E}(B_{ij})$ is nearly zero. We define the informal idealized quantities $\widetilde{\mathbf{B}}$ and $\widetilde{Q}_{XY}$ in comparison with Eqs. (7) and (8):

$$\widetilde{\mathbf{B}} = \mathbf{A} - \frac{1}{\pi^{++}} \boldsymbol{\pi}^{(X)} (\boldsymbol{\pi}^{(Y)})^{\top}, \tag{11}$$

and

$$\widetilde{Q}_{XY}(\hat{g}^{(X)}, \hat{g}^{(Y)}) = \frac{1}{\pi^{++}} \sum_{i \in \hat{g}^{(X)}} \sum_{j \in \hat{g}^{(Y)}} \widetilde{B}_{ij}, \tag{12}$$

where the empirical degree distributions $\mathbf{d}^{(X)}$ and $\mathbf{d}^{(Y)}$ have been replaced by the theoretical node connectivity parameters $\boldsymbol{\pi}^{(X)}$ and $\boldsymbol{\pi}^{(Y)}$, and the empirical normalization factor d^{++} is also replaced by the theoretical normalization factor π^{++}.

If the pairing of X- and Y-node groupings $\hat{g}^{(X)}$ and $\hat{g}^{(Y)}$ exhibit some co-community structure, then Eq. (10) no longer holds, and so the null model does not hold either. The stronger this co-community structure is, the further we move from the null model, and the greater θ becomes relative to $1/\pi^{++}$. This corresponds to $\mathbb{E}(A_{ij})$ becoming larger than $\pi_i^{(X)} \pi_j^{(Y)} / \pi^{++}$, which is equivalent to the observed number of edges in the co-community becoming greater than the expected, under the null of no co-community structure. This in turn means that

$\widetilde{Q}_{XY}$ also becomes more positive. In other words, the further we move from the null model, the greater tendency of the X nodes and Y nodes of these groups to form connections with one another (compared with their expected propensity to make connections with any nodes, of the opposite type) and therefore constitute a strong co-community. Hence, a parsimonious method of comparing potential co-communities is simply to compare their local comodularity, $Q_{XY}(\hat{g}^{(X)}, \hat{g}^{(Y)})$. This naturally leads to a ranking of potential co-communities according to their strength, e.g., leading to identification of the shaded blocks in Fig. 1 by some thresholding criterion, such as statistical significance.

An estimate of statistical significance of a potential co-community can also be made, as follows. Noting that, with adjacency matrix $\mathbf{A}$ defined according to the Bernoulli distribution of Definition 1, with fixed $\theta_{z^{(X)}(i),z^{(Y)}(j)} = 1/\pi^{++}$,

$$\mathrm{Var}(\widetilde{B}_{ij}) = \mathrm{Var}(A_{ij}) = \left(\frac{\pi^{(X)}\pi^{(Y)}}{\pi^{++}}\right)\left(1 - \frac{\pi^{(X)}\pi^{(Y)}}{\pi^{++}}\right),$$

and assuming probabilities of observing links between different pairs of nodes are independent, the variance of $\widetilde{Q}_{XY}(\hat{g}^{(X)}, \hat{g}^{(Y)})$ can be approximated (for deterministic node-groupings) as

$$\mathrm{Var}(\widetilde{Q}_{XY}(\hat{g}^{(X)}, \hat{g}^{(Y)})) = \frac{1}{(\pi^{++})^2}\sum_{i\in\hat{g}^{(X)}}\sum_{j\in\hat{g}^{(Y)}}\left(\frac{\pi_i^{(X)}\pi_j^{(Y)}}{\pi^{++}}\right)\left(1 - \frac{\pi_i^{(X)}\pi_j^{(Y)}}{\pi^{++}}\right), \tag{13}$$

where the factor $1/(\pi^{++})^2$ is due to the factor $1/(\pi^{++})$ in Eq. (12). We also note that here, departing from convention, the $\pi_i^{(X)}$ and $\pi_j^{(Y)}$ are of the scale of degrees rather than probabilities (Definition 1). Hence, assuming $\mathbf{d}^{(X)} \xrightarrow{p} \boldsymbol{\pi}^{(X)}$, $\mathbf{d}^{(Y)} \xrightarrow{p} \boldsymbol{\pi}^{(Y)}$, and $d^{++} \xrightarrow{p} \pi^{++}$, and assuming the potential co-community defined by $\hat{g}^{(X)}$ and $\hat{g}^{(Y)}$ is comprised of sufficiently many nodes for a Gaussian approximation to hold, we can test the significance of $Q_{XY}(\hat{g}^{(X)}, \hat{g}^{(Y)})$ with a z test, with zero mean and with $\mathrm{Var}(Q_{XY})$ estimated as $\mathrm{Var}(\widetilde{Q}_{XY})$ in Eq. (13), also replacing $\pi_i^{(X)}$ with $d_i^{(X)}$, $\pi_j^{(Y)}$ with $d_j^{(Y)}$, and π^{++} with d^{++}, and ignoring the stochasticity of $g^{(X)}$ and $g^{(Y)}$. A pairing $\hat{g}_p^{(X)}$ and $\hat{g}_q^{(Y)}$ is then defined as a co-community $\hat{c}$ and included in $\hat{C}$ (Definition 3), i.e., $\{p, q\} = \hat{c} \in \hat{C}$, if and only if this pairing $\hat{g}_p^{(X)}$ with $\hat{g}_q^{(Y)}$ is significant according to this z test, at some significance level. We note that, in practice, this is only a rough approximation of significance, also because by specifying in advance the co-community node groupings $\hat{g}^{(X)}$ and $\hat{g}^{(Y)}$, we have introduced dependencies between the X and Y nodes of this co-community.

B. Arranging the co-communities for visualization

A standard task in exploratory data analysis using variants of the stochastic blockmodel is arranging the detected communities so they can be visualized in a helpful way. This visualization is usually carried out by way of a heatmap representation of the adjacency matrix with the nodes grouped into communities. In the symmetric or unipartite community detection scenario, the communities occur along the leading diagonal of this ordered adjacency matrix. The communities themselves are often ordered along the leading diagonal according to their edge densities. In the bipartite co-community detection setting, co-communities may be present away from the leading diagonal, and there is no longer a restriction on how many co-communities a node may be part of—although we do not consider here the possibility of overlapping co-communities.

We propose then, that once the X-node groupings and Y-node groupings have been determined by spectral clustering as described above, a natural way to order these groups with respect to one another is via row and column comodularities, which we define as follows.

Definition 5. With d^{++} given by Definition 3, and with $\mathbf{B}$ given by Definition 4, with with the set of X-node groupings and the set of Y-node groupings estimated according to Algorithm 1 as $\hat{G}^{(X)}$ and $\hat{G}^{(Y)}$, respectively, the row and column modularities $Q_{\mathrm{row}}(\hat{g}^{(X)})$ and $Q_{\mathrm{column}}(\hat{g}^{(Y)})$ are defined, for $\hat{g}^{(X)} \in \hat{G}^{(X)}$ and $\hat{g}^{(Y)} \in \hat{G}^{(Y)}$, as

$$Q_{\mathrm{row}}(\hat{g}^{(X)}) = \sum_{\hat{g}^{(Y)}\in\hat{G}^{(Y)}}\left|\frac{1}{d^{++}}\sum_{i\in\hat{g}^{(X)}}\sum_{j\in\hat{g}^{(Y)}}B_{ij}\right| \tag{14}$$

and

$$Q_{\mathrm{column}}(\hat{g}^{(Y)}) = \sum_{\hat{g}^{(X)}\in\hat{G}^{(X)}}\left|\frac{1}{d^{++}}\sum_{i\in\hat{g}^{(X)}}\sum_{j\in\hat{g}^{(Y)}}B_{ij}\right|. \tag{15}$$

Considering the absolute values of the local comodularities in these sums serves to prioritize the most extreme choices of divisions of nodes into co-communities, according to their local comodularities. On the other hand, if absolute values were not considered here, the row and column modularities would always be zero, because the rows and columns of $\mathbf{B}$ must always sum to zero. We note that we could have chosen to use squared instead of absolute values; however, we choose to use absolute values so as not to give extra weight to a few extreme values. The row and column comodularities are the sums, respectively, of the absolute values of the local comodularities along the rows and columns, respectively, of the ordered adjacency matrix. Hence, they represent a measure of how extreme the co-community divisions are, in each row and column, according to the groupings defined by $\hat{G}^{(X)}$ and $\hat{G}^{(Y)}$. By ordering the X-node and Y-node groupings by decreasing $Q_{\mathrm{row}}(\hat{g}^{(X)})$ and $Q_{\mathrm{column}}(\hat{g}^{(Y)})$, respectively, co-communities with the largest local comodularities will tend to congregate towards the top left of the ordered adjacency matrix. This is a natural arrangement for visualization as a heatmap, because it tends to place the strongest co-communities together in this corner, and so the attention is intuitively drawn to this region.

We note that there may be other equally effective ways of arranging the adjacency matrix for visualization as a heatmap. However, this method is effective, and it is a parsimonious solution in the context of comodularity, because row and column modularities are very simply and intuitively related to local comodularity. In the case that there is no co-community structure present, such as under the null model of Eq. (9), then Q_{row} and Q_{column} as defined in Definition 5 would also tend to be close to zero, and the ordering would cease to be meaningful. However, if there are even a few significant co-communities

present, their corresponding X- and Y-node groupings $\hat{g}^{(X)}$ and $\hat{g}^{(Y)}$ would stand out, as assessed by Q_{row} and Q_{column}. Therefore these $\hat{g}^{(X)}$ and $\hat{g}^{(Y)}$ would be placed at the top of the respective orderings, with the co-community pairings tending towards in the top-left corner. The other rows and columns, which do not contain significant co-communities, would have corresponding Q_{row} and Q_{column} close to zero. Hence, these rows and columns would be naturally ordered according to their irrelevance. They would accordingly be placed further away from the top left of the heatmap, giving the intuition that they are unimportant.

C. Defining an objective function for optimizing the co-community partitions

Defining an objective function over the whole network, in terms of the assignments of the nodes to X-node and Y-node groupings $\hat{g}^{(X)}$ and $\hat{g}^{(Y)}$, allows optimization of these node assignments. It also provides a means of comparison of algorithmic parameters and other design choices in the practical implementation of the methods. It would be most ideal, for a trial assignment of nodes to $\hat{G}^{(X)}$ and $\hat{G}^{(Y)}$, to estimate the set of co-communities $\hat{C}$ using the method of Sec. III A, and then to calculate the comodularity according to Definition 3. However, for a large number of repetitions within an algorithm, or for an iterative search and optimization, this would be computationally inefficient. Instead, we define the global comodularity to be used as an objective function for such purposes, as follows:

Definition 6. With d^{++} given by Definition 3, and with $\mathbf{B}$ given by Definition 4, with the set of X-node groupings and the set of Y-node groupings estimated according to Algorithm 1 as $\hat{G}^{(X)}$ and $\hat{G}^{(Y)}$, respectively, the global comodularity is defined, for $\hat{g}^{(X)} \in \hat{G}^{(X)}$ and $\hat{g}^{(Y)} \in \hat{G}^{(Y)}$, as

$$Q_{\text{global}} = \sum_{\hat{g}^{(Y)} \in \hat{G}^{(Y)}} \sum_{\hat{g}^{(X)} \in \hat{G}^{(X)}} \left| \frac{1}{d^{++}} \sum_{i \in \hat{g}^{(X)}} \sum_{j \in \hat{g}^{(Y)}} B_{ij} \right|. \tag{16}$$

For a pairing $\hat{g}^{(X)}$ and $\hat{g}^{(Y)}$, the local comodularity $Q_{XY}(\hat{g}^{(X)}, \hat{g}^{(Y)})$ represents the strength of the co-community structure in that grouping of X nodes and Y nodes. If the absolute value was not considered in the sum, Q_{global} would always be zero. Hence, by prioritizing a sum of the absolute values of the local comodularity of all pairings $\hat{g}^{(X)}$ with $\hat{g}^{(Y)}$, we prioritize an extreme division of the X nodes and Y nodes into co-communities, as measured by the local comodularity. This therefore corresponds to an extreme partition in terms of co-community structure, as assessed by comodularity.

Spectral clustering usually requires the nodes to be grouped in the spaces of the top singular vectors of the co-Laplacian, and this grouping is often carried out by k means, as described in Algorithm 1. Because k-means optimization is not convex, the converged result may be a local optimum. Hence, implementations of k means often begin at a random start point, with the optimization run several times from random start points, choosing the result which is in some sense optimal. In the community-detection setting, a natural statistic to maximize in this optimization is the Newman-Girvan modularity. An equivalent statistic here to maximize in this co-community detection setting is hence the global comodularity, which is intuitively linked to the local comodularity measure of co-community structure. In the community-detection setting, assignments to communities can also be optimized by carrying out node swapping between communities, in order to maximize the Newman-Girvan modularity [37]. The global comodularity is a statistic which could be equivalently maximized in this co-community detection setting.

IV. EXAMPLES

In this section, we present results of applying the proposed methodology to simulated data, and to real data relating to movie reviews and to genomic patterns. We fit the degree-corrected stochastic coblockmodel by spectral clustering as described in Sec. II, with additional practical details as described below. As noted in Sec. I, methodology to determine the optimal number of clusters is a challenging problem, worth studying independently of methodology to determine the clusters themselves [21]. Therefore to enable us to properly evaluate our proposed methodology for detecting co-communities (as illustrated by the shaded blocks in Fig. 1), we use the ground-truth cluster numbers defined in the simulation study to focus on testing whether we can recover the planted co-communities. Then in Sec. IV C, we are limited practically in how many clusters we should aim to detect by the granularity of the ground truth (i.e., node covariate information) that is available. Again, our primary aim is to assess the proposed methodology for detecting co-communities: in this example, a co-community represents a group of similar movie fans who like a set of similar movies. We also note that Appendix C provides a practical method with theoretical justification for estimating the optimal number of X- and Y-node groupings $k^{(X)}$ and $k^{(Y)}$, from which co-communities can be identified. In the context of community detection, fitting the degree-corrected stochastic blockmodel using spectral clustering, when calculating the Laplacian it is advantageous to slightly inflate the degree distribution (regularization) [4], a trick which made Google's original page-rank algorithm [38] so effective in web searching. Here in the co-community detection setting, correspondingly when calculating the co-Laplacian [Eq. (6)], we inflate the diagonals of $\mathbf{D}^{(X)}$ and $\mathbf{D}^{(Y)}$ by the medians of $\mathbf{d}^{(X)}$ and $\mathbf{d}^{(Y)}$, respectively. Further, when fitting variants of the stochastic blockmodel by spectral clustering with k means, nodes with small leverage score (which are usually low-degree nodes) can be excluded from the k-means step [4]; this practice is also followed here. We note that these regularization steps have not previously been carried out in this co-community detection and/or coclustering setting. We also note that while spectral clustering is in general computationally intensive, binary adjacency matrices such as those dealt with in this setting tend to be very sparse. Further, we only require $k = \text{Max}(k^{(X)}, k^{(Y)})$ components of the singular value decomposition, a number which tends to be two or more orders of magnitude smaller than the maximum dimension of the adjacency matrix. Efficient computational methods exist to find the top few components in the singular value decomposition of large sparse matrices [39,40], with implementations in MATLAB and R, meaning that these methods are easy to implement and practical for large networks. The k-means clustering algorithm begins with a random start point,

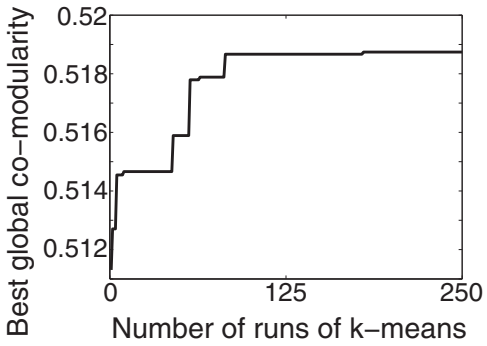

FIG. 2. Convergence of the comodularity. The comodularity converges well to a maximum, within 250 runs of k means, in the genomics data set. For reference, the comodularity is consistently found to be zero when calculated based on randomly assigned co-community partitions of similar size.

and hence it can provide a different result each time it is run. We therefore run the k-means step in the spectral clustering several times, choosing the result which maximizes the global comodularity [Eq. (16)]. We run k means repeatedly until the output is visually assessed to have stabilized, at which point it can be seen from the convergence plot that there is very little, if any, improvement in comodularity achieved by further runs of k means. An example of such convergence in the genomics data set presented in Sec. IV B is shown in Fig. 2.

A. Simulation study

We carried out a simulation study to evaluate the effectiveness of this co-community detection methodology against generated networks with known ground-truth co-communities. A classic generative model for exchangeable random networks with heterogeneous degrees is the logistic-linear model [34]. We use a version here for bipartite networks, with additional co-community structure, defined as

$$\text{Logit}(p_{ij}) = \alpha_i^{(X)} + \alpha_j^{(Y)} + \theta_{ij}, \tag{17}$$

where p_{ij} defines the probability of an edge being observed between nodes i and j. We choose to use this model because the parameters can take any real values, and the edge probabilities p_{ij} will still be between zero and 1. This model only deviates from the equivalent logarithmic model when the parameter values become very large, which is what prevents p_{ij} from reaching (and exceeding) 1 [34]. Further, the blockmodel approximates any smooth function, and hence the model can be used purely in the sense of approximation [16,23]. The node-specific parameters $\alpha_i^{(X)}$ and $\alpha_j^{(Y)}$ are elements of parameter vectors $\boldsymbol{\alpha}^{(X)}$ and $\boldsymbol{\alpha}^{(Y)}$ which define degree distributions for the X and Y nodes. We choose power-law degree distributions for the nodes, because this is a characteristic of scale-free networks [41], which are found to be physically realistic in a wide range of scenarios, including biological networks [42] and social networks [43]. We note that although power-law degree distributions are not found in exchangeable networks, blockmodels (such as the model presented here) are still good at approximating the propensity for connections within the network [44]. The parameters $\alpha_i^{(X)}$ and $\alpha_j^{(Y)}$ are each generated as the logarithms of samples taken from a bounded Pareto distribution as in [45]. We note that because $\alpha_i^{(X)}$ and $\alpha_j^{(Y)}$ are chosen to be random, our generated networks are exchangeable [46], whereas if $\alpha_i^{(X)}$ and $\alpha_j^{(Y)}$ were defined deterministically, these networks would instead be generated under the inhomogeneous random graph model [47,48]. The co-community parameter θ_{ij} is allowed to take two values: $\theta_{ij} = \theta_{\text{in}}$ if i and j are in the same co-community, and $\theta_{ij} = \theta_{\text{out}}$ otherwise, which is equivalent to the modeling constraint we applied in Eq. (2). After generating the p_{ij} according to Eq. (17), the network is generated by sampling each A_{ij},

$$A_{ij} \sim \text{Bernouilli}(p_{ij}).$$

The co-communities themselves are planted in the network as randomly chosen groups of 150 of each type of node. We note that some nodes may not lie in any co-community: for nodes not in a co-community, the edge probability is regulated by θ_{out}, and similarly by θ_{in} for two nodes that are in a co-community. The maximum number of co-communities is set at $k^{(X)} \times k^{(Y)}$, and by analogy with the unipartite or symmetric community detection setting, we choose to set the number of co-communities T as the square root of this theoretical maximum, $T = \sqrt{k^{(X)} \times k^{(Y)}}$. As discussed in Sec. III, in the unipartite community detection setting there is a constraint on the number of communities, $k = k^{(X)} = k^{(Y)}$, because the X-node and Y-node groupings are the same thing. This constraint does not exist in the bipartite co-community detection setting, and so the theoretical maximum number of co-communities is $k^{(X)} \times k^{(Y)}$, i.e., the square of the number of communities in the equivalent symmetric community detection setting. However, we expect the number of co-communities to be significantly less than this in practice, and so by default we choose $T = \sqrt{k^{(X)} \times k^{(Y)}}$ as the number of co-communities, although we note that many other choices would also be valid here.

We test the methods on networks generated with $k^{(X)}$ and $k^{(Y)}$ ranging from 8 and 6, respectively, up to 80 and 60, respectively (corresponding to values of numbers of nodes, m and l, ranging from 1200 and 900 up to 12 000 and 9000, respectively). We also test the methods on networks generated with values of θ_{in} from 10 to 50, which corresponds to within co-community edge density $\rho_{\text{in}} \in \{0.039, 0.15, 0.34, 0.6\}$, and we set $\theta_{\text{out}} = 1$, corresponding to outside or between co-community edge density $\rho_{\text{in}} = 0.0013$ [N.B., θ_{in} and θ_{out} are not probabilities; see Eq. (17)]. For each combination of parameters, we carry out 50 repetitions of network generation and co-community detection, to enable assessment of the variability of the accuracy of the co-community detection (with more repetitions, the computational cost becomes prohibitive). After generating the networks, we detect co-communities according to the methods described above, based on the same values of $k^{(X)}$ and $k^{(Y)}$ that we used to generate the networks. We keep these values the same, to understand specifically how the co-community detection methodology is working, as discussed in more detail in Sec. I. This means there are $k^{(X)} \times k^{(Y)}$ potential co-communities, and we assess each in terms of strength and significance, as discussed in Sec. III A. Hence, we define the estimated set of co-communities $\hat{C}$, as all combinations of detected X- and Y-node groupings $\hat{g}^{(X)} \in \hat{G}^{(X)}$ with $\hat{g}^{(Y)} \in \hat{G}^{(Y)}$

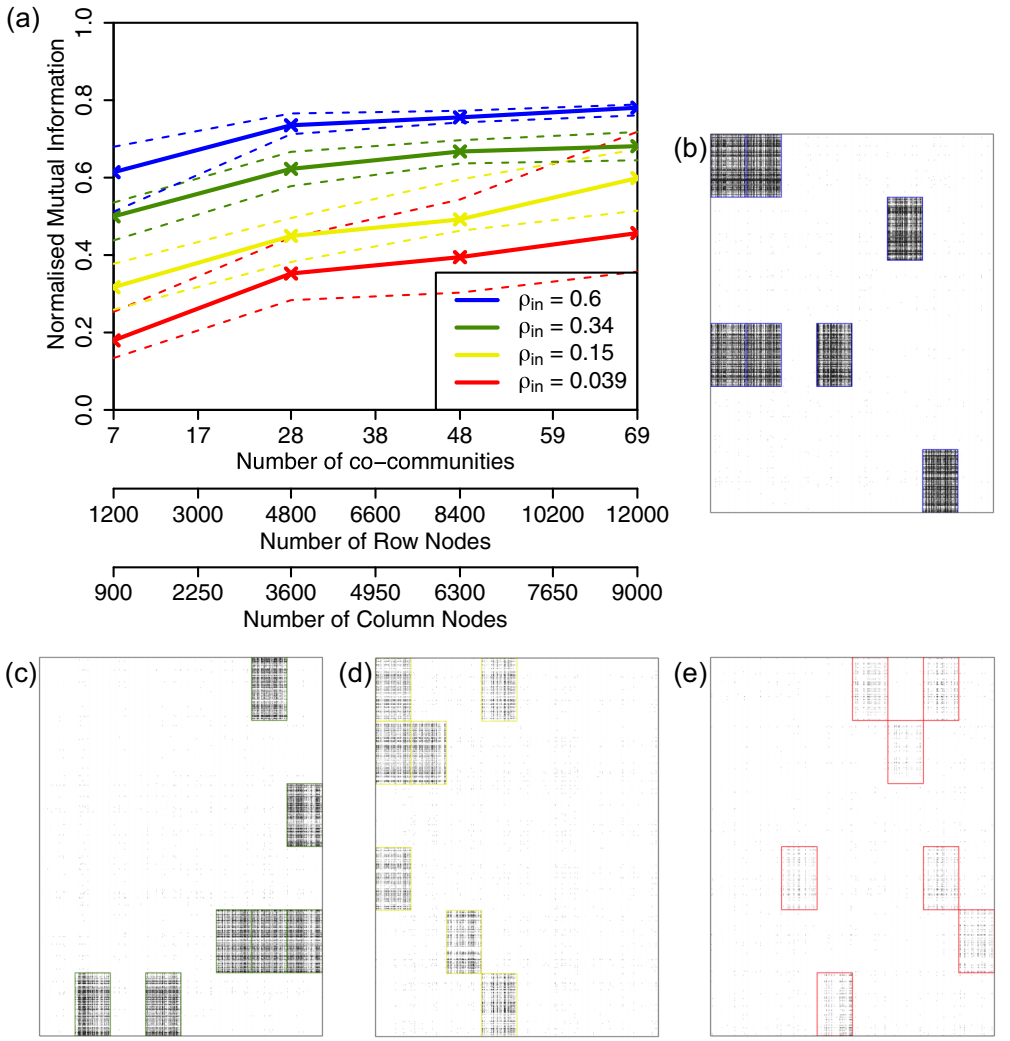

FIG. 3. Simulation study. (a) Normalized mutual information (NMI) compares detected co-communities with ground-truth planted co-communities (dashed lines indicate 95% C.I.). (b)–(e) Examples of generated networks all with $nR = 1200$, $nC = 900$, $kR = 8$, $kC = 6$, and 7 planted co-communities; entries in the adjacency matrix equal to 1 (representing a network edge) are marked in black, and planted co-communities are outlined in color. (b) $\theta_{\text{in}} = 40$, within-community edge density $\rho_{\text{in}} = 0.6$; (c) $\theta_{\text{in}} = 30$, $\rho_{\text{in}} = 0.34$; (d) $\theta_{\text{in}} = 20$, $\rho_{\text{in}} = 0.15$; (e) $\theta_{\text{in}} = 10$, $\rho_{\text{in}} = 0.039$. For all networks, $\theta_{\text{out}} = 1$, outside (between) co-community edge density $\rho_{\text{out}} = 0.0013$.

which are significant according to a z test with zero mean and variance calculated as in Eq. (13). We define significance according to false discovery rate (FDR)-corrected [49] p-value < 0.05. This tends to result in more co-communities being detected than were originally planted (primarily due to some being split); however, we note that the main aim of this methodology is to find a good representation of the underlying co-community structure (as assessed by comodularity), rather than to reproduce it exactly.

To compare detected co-communities with the ground-truth planted co-communities, we use the normalized mutual information (NMI) [50] to compare the corresponding X- and Y-node groupings (over the full node sets). The NMI compares the numbers of nodes which appear together in the discovered X- and Y-node groupings, compared with whether they appeared together in those that correspond to the planted co-communities (adjusted for group sizes). NMI has been used in a similar way previously in the co-community-detection context [51], as well as the unipartite community-detection context [11]. The NMI takes the value 1 if the X- and Y-node groupings that correspond to these co-communities are perfectly reproduced in the co-community detection, and zero if they are not reproduced at all, and somewhere in between if they are partially reproduced. The results, together with examples of randomly generated adjacency matrices, are shown in Fig. 3, which shows that the method performs well as long as there is sufficient within-co-community edge density (implying a detection threshold), and performs well as the number of co-communities increases. The run times for the fits for number of row nodes equal to 1200, 4800, 8400, and 12 000 (with number of column nodes equal to 900, 3600, 8400, and 9000) are 1.6, 21, 80, and 190 s, respectively (using one core of a MacBook Pro, 2019, 2.6 GHz).

B. Genomics data set

We present an example of a practical application of these methods to a challenging problem in genomics. A gene

encodes how to make a gene product, and the corresponding gene expression level quantifies how much gene product is currently being produced. Hence, the gene expression level indicates the extent to which a gene is "active," or "switched on." DNA methylation is a gene regulatory pattern, meaning that it influences the activity and/or expression level of particular genes. DNA methylation patterns are themselves influenced by the expression levels and/or activity of other genes. However, much is still unknown about the interaction between DNA methylation patterns and gene expression patterns [52]. It is of much interest to uncover groups of genes with methylation patterns which are linked to the expression patterns of other groups of genes, to allow biological hypotheses to be formed, which can then be investigated further, experimentally and computationally. Hence, this is a natural scenario to be approached with co-community detection, as the method offers the potential to uncover latent structure not easily identifiable otherwise.

As a measure of the DNA methylation (DNAm) pattern of each gene, we choose to consider here intragene DNA methylation variability (IGV), as it is a per-gene measure of DNA methylation variance which has been shown to be strongly associated with disease [53,54]. We denote the gene expression variables $X(i)$, $i = 1, \ldots, m$, and the DNAm variables $Y(j)$, $j = 1, \ldots, l$; i.e., $X(i)$ and $Y(j)$ refer to the measurements for particular genes of gene expression and DNA methylation IGV, respectively. We define a network edge $A_{ij} = 1$ if variables $X(i)$ and $Y(i)$ are significantly correlated, and we set $A_{ij} = 0$ otherwise [55]. We carried out co-community detection on this genomics data set according to the methods described above. (The data source is the Cancer Genome Atlas [56], breast cancer invasive carcinoma data set, basal tumor samples only). Figure 4(a) shows the adjacency matrix after carrying out co-community detection, ordering the X- and Y-node groupings by row and column comodularity [Eqs. (14) and (15)]. Figure 4(b) (inset) shows the same adjacency matrix ordered along its margins alphabetically by gene name, i.e., without ordering the margins using co-community detection. Hence, Fig. 4(b) shows a baseline in which the nodes are essentially randomly ordered, against which to compare the adjacency matrix after co-community detection, and ordering based upon it. The co-community structure is clearly revealed in Fig. 4(a), whereas no co-community structure is visible in Fig. 4(b). We define a co-community $\hat{c} \in \hat{C}$ as a combination of X-node grouping $\hat{g}^{(X)} \in \hat{G}^{(X)}$ with Y-node grouping $\hat{g}^{(Y)} \in \hat{G}^{(Y)}$ which is significant according to a z test with zero mean and variance calculated as in Eq. (13), with significance defined by FDR-corrected p-value < 0.05. The numbers of X- and Y-node groupings, $k^{(X)}$ and $k^{(Y)}$, are estimated according to Eqs. (C6) and (C7) as 89 and 67, respectively, leading to 5963 potential co-communities, of which $\hat{T} = 2018$ are found to be significant. We tested these 2018 significant co-communities for domain relevance, by comparing the overlap of the genes (nodes) of each co-community, separately with each of 10 295 known gene groups [58]. This type of analysis is often called "gene set enrichment analysis" (GSEA) [59]. We found that 1340 (66%) overlap significantly (Fisher's exact test, FDR-adjusted $p < 0.05$) with these known gene sets, confirming the domain relevance of this result, as well as indicating novel findings which could be investigated further by experimental biologists.

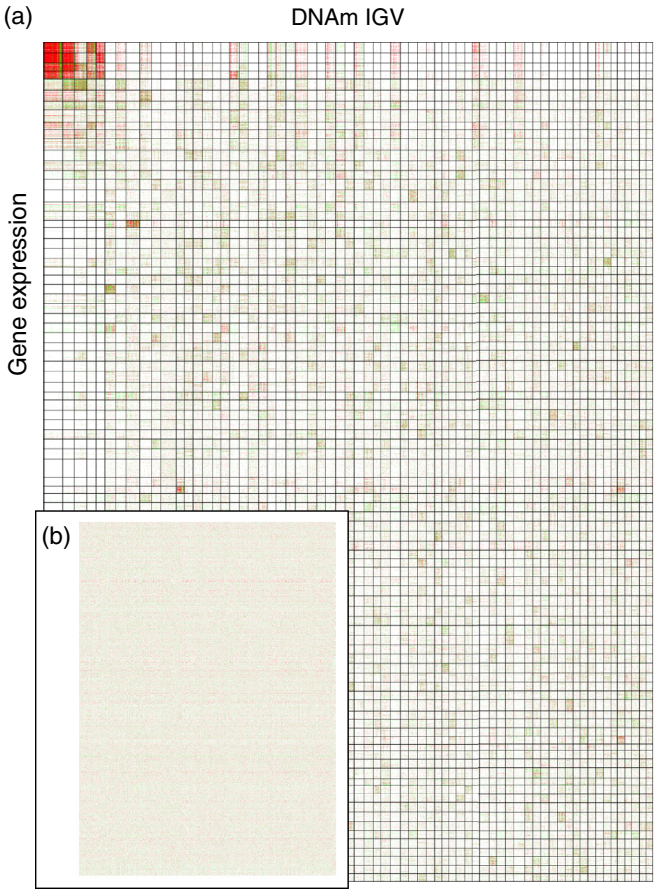

FIG. 4. Co-Communities in the genomics data set. (a) Genes are ordered along the margins of the adjacency matrix, according to co-communities detected by the methods presented here. Partitions between detected co-communities are shown with black lines. (b) The same adjacency matrix ordered along its margins alphabetically by gene name, i.e., without ordering the margins using co-community detection. Entries in the adjacency matrix equal to 1 (representing a network edge) are colored, with green and red indicating positive and negative associations, respectively. We note that this is a signed network [57]. N.B., this color scheme contrasts with that used in Fig. 5, in which network edges do not have associated signs, and hence are all colored blue.

C. Movie-review data set

We present a second, contrasting example of a practical application of these methods to real data, to a consumer-product review data set. We downloaded movie review data from the Movie Lens database, which details 1 000 209 reviews of 3952 different movies, by 6040 unique users who each provided at least 20 different reviews [60]. Denoting movies by the variables $X(i)$, $i = 1, \ldots, m$, and users by the variables $Y(j)$, $j = 1, \ldots, l$, we define a network edge, i.e., $A_{ij} = 1$, if movie $X(i)$ has been reviewed by user $Y(i)$, and no edge, i.e., $A_{ij} = 0$, otherwise. Covariate information is also available, assigning each movie to one of 18 categories, and classifying each user into one of 7 age groups and 20 professions.

As discussed in Sec. I, determining the optimal number of clusters is a different problem than determining the clusters themselves. In this example, the granularity of the available ground-truth clusters (i.e., the covariate information we have

available for verification of detected coclusters) is very much less than that estimated according to Eqs. (C6) and (C7) (i.e., 77 and 95, respectively). To make the comparison with the available ground truth straightforward, we choose to use a level of granularity of $k^{(X)} = 10$ and $k^{(Y)} = 15$ that is in line with the available ground truth. This is well justified theoretically, as follows. The graphon function [61] regulates the smoothness of the success probabilities generating the edges of the network, and so if the success probabilities are changing rapidly across nodes, then we need to use more blocks, or communities; in doing so, we are ensuring that even the roughest portion of the graphon function is sufficiently well resolved. However, we can also reduce granularity by reducing the number of blocks, while accepting that we will lose the ability to resolve some portions of the graphon function well. This point is also discussed further in Appendix C.

Co-Community detection was carried out using the methods described above, and Fig. 5 shows the result. Of the 150 potential co-communities, $\hat{T} = 41$ are found to be significant using the z test as before, and are thus defined as co-communities. The X and Y nodes of these 41 co-communities are tested for overlap with the covariate groups. Of these, 22 are found to overlap significantly with one or more covariate groups, and these are highlighted in red, with the significant covariate groups shown along the margins, in Fig. 5. We use Fisher's exact test to assess significances, defined as FDR-corrected $p < 0.05$. We note that using the Fisher test is slightly inappropriate due to the generative mechanism of the groups. However, we would anticipate that any residual correlation from the group discovery is minor. Importantly though for assessment via this benchmark data set, many of the findings are predictable: horror, science fiction,i and war films tend to be watched by younger people; drama and romance are popular across the board. Others need more explanation, for example, a group of children's movies and musicals tended to be reviewed by 25–34-year-old customer service professionals. However, we can expect that this is a demographic group of people who tend to have younger children whom they watch movies with. Other children's movies are grouped together with animation, fantasy, and horror, and tend to be watched by both younger and older groups. This might reflect very broad classifications used for such movies, many of which in reality could be fairly similar. Also these are groups of people who would tend to watch movies together. An important conclusion to draw is that the covariate information available for this data set appears to be of a lower granularity than the detail which can be revealed by these co-community detection methods.

V. CONCLUSION

We have introduced the notion of comodularity based on the stochastic coblockmodel, and have shown how it can be used to perform co-community detection in bipartite networks. We have shown how comodularity can be used to compare co-communities, to calculate their strength and significance, and to arrange them for visualization. We have addressed practical points about the implementation of the methodology, and have demonstrated its usefulness with a simulation study and application to two contrasting examples of real data sets, from genomics and consumer-product reviews. We note that the main aim of our method is detection of co-communities (in fact misspecified, because we do not think that the inferred groups are perfect). An interesting extension to this methodology would be to consider overlapping blocks in the stochastic coblockmodel, a problem which has already been successfully addressed in the context of the stochastic blockmodel for unipartite networks [62], and in coclustering without fitting the stochastic blockmodel [24]. Another interesting application would be to develop an online version of the method (i.e., which updates rather than recomputes) as a computationally efficient approach to large and growing data sets [63]. This methodology would be expected to work similarly well in many other contexts, such as interpersonal networks where the individuals are of two distinct categories, such as teachers and students, or publication networks where the two types of variables are authors and papers. This methodology could also be expected to work in even more general settings of biclustering or coclustering, in which the variables being clustered together are simply correlated, rather than having any tangible interactive behavior in the real world. These methods are based on commonly available computationally efficient methods for large sparse matrices, and perform well on large data sets, with large numbers of co-communities, often performing better than methods based on model likelihoods.

ACKNOWLEDGMENT

The procedure in Appendix C for estimating optimal numbers of X-node and Y-node groupings was developed as part of the author's Ph.D. thesis under the guidance of Prof. Sofia Olhede.

APPENDIX A: DERIVATION RELATING TO ALGORITHM 1, FOR THE CASE OF TWO CO-COMMUNITIES

Define m, l, $\mathbf{A}$, $\mathbf{B}$, $\mathbf{d}^{(X)}$, $\mathbf{d}^{(Y)}$, d^{++}, $g^{(X)}$, $g^{(Y)}$, $k^{(X)}$, $k^{(Y)}$, Ψ, and Q_{XY} according to Definitions 1–4. Specify that $k^{(X)} = k^{(Y)} = 2$, that $T = 2$, that $c_1 = \{1, 1\}$, and that $c_2 = \{2, 2\}$; i.e., that there are two co-communities, the first of which consists of $g_1^{(X)}$ paired with $g_1^{(Y)}$, and the second of which consists of $g_2^{(X)}$ paired with $g_2^{(Y)}$. Define co-community label vectors $\mathbf{s}$ and $\mathbf{r}$ for the X and Y nodes, respectively, such that

$$s_i = \begin{cases} 1 & \text{if } X\text{-node } i \text{ is in co-community 1} \\ -1 & \text{if } X\text{-node } i \text{ is in co-community 2,} \end{cases} \tag{A1}$$

and

$$r_j = \begin{cases} 1 & \text{if } Y\text{-node } j \text{ is in co-community 1} \\ -1 & \text{if } Y\text{-node } j \text{ is in co-community 2.} \end{cases} \tag{A2}$$

Hence (referring to Definition 3),

$$\Psi(C; G^{(X)}, G^{(Y)}; i, j) = \tfrac{1}{2}(s_i r_j + 1),$$

and

$$Q_{XY} = \frac{1}{2d^{++}} \sum_{i=1}^{m} \sum_{j=1}^{l} B_{ij}(s_i r_j + 1).$$

Note that the rows of $\mathbf{B}$ sum to zero:

$$\sum_{j=1}^{l} B_{ij} = \sum_{j=1}^{l} A_{ij} - \frac{d_i^{(X)}}{d^{++}} \sum_{j=1}^{l} d_j^{(Y)} = d_i^{(X)} - \frac{d_i^{(X)}}{d^{++}} d^{++} = 0.$$

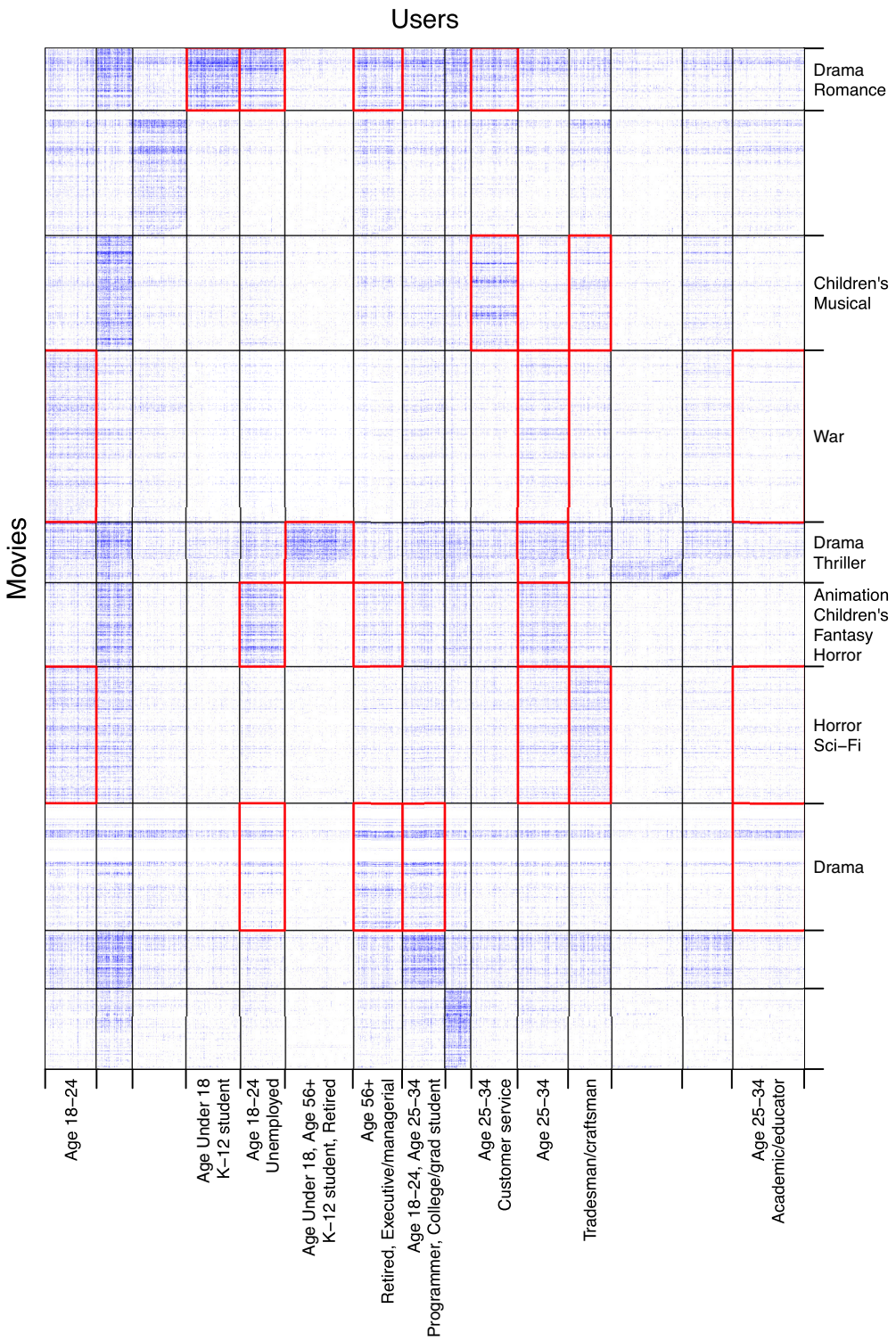

FIG. 5. Co-Communities in the movie-review data set. Entries in the adjacency matrix equal to 1 (representing a network edge) are colored blue, and detected communities are outlined in black.

Also, the columns of **B** also sum to zero, by a similar argument. Hence,

$$Q_{XY} = \frac{1}{2d^{++}} \sum_{i=1}^{m} \sum_{j=1}^{l} B_{ij} s_i r_j. \tag{A3}$$

When Newman [33] derives the properties of unipartite network community detection he relaxes the constraint that the co-community labels take the values of ± 1, to be able to arrive at an algebraic solution. Nodes are then assigned to one community or the other, according to their sign (in the two-community scenario). A similar relaxation is made

here, allowing $s_i \in \mathbb{R}$ and $r_j \in \mathbb{R}$, subject also to the following elliptical constraints, which allow for degree heterogeneity as in the degree-corrected stochastic blockmodel:

$$\sum_{i=1}^{m} d_i^{(X)} s_i^2 = d^{++}, \tag{A4}$$

$$\sum_{j=1}^{l} d_j^{(Y)} r_j^2 = d^{++}. \tag{A5}$$

In the extreme scenario, in which $s_i \in \{-1, 1\}$ and $r_j \in \{-1, 1\}$, these constraints are equivalent to $d^{++} = \sum_{i=1}^{m} d_i^{(X)} = \sum_{j=1}^{l} d_j^{(Y)}$ (i.e., as per Definition 3). This relaxation is equivalent to saying that nodes may be partly in one group, and partly in another group (which also relates to the mixed-membership blockmodel [12]). N.B., ultimately each node will be assigned entirely to only the group it is most strongly associated with (according to s_i or r_j), and hence mixed membership does not occur in the final assignment of nodes to groups. For homogeneous degree distributions, the constraints of Eqs. (A4) and (A5) prevent the comodularity from becoming arbitrarily large, as nodes are assigned many times over to many groups. For heterogenous degree distributions, the effect of the constraint is equivalent, except that the constraint is weighted to give importance to high-degree nodes. This is achieved by the constraints of Eqs. (A4) and (A5) restricting the weighted sum of the degrees (weighted by the assignment of nodes to groups) to be equal to the total number of edges.

We wish to find the community assignment vectors $\mathbf{r}$ and $\mathbf{s}$ which maximize the comodularity; i.e., we want to maximize Q_{XY} with respect to both $\mathbf{r}$ and $\mathbf{s}$. To do this, we employ the Lagrange multipliers λ and μ, and equate the derivatives to zero; N.B., the partial derivatives with respect to $s_{i'}$ and $r_{j'}$ are used as the derivatives and are taken with respect to these individual $i' \in \{1, \ldots, l\}$, and $j' \in \{1, \ldots, m\}$.

$$\frac{\partial}{\partial s_{i'}}\left[\sum_{i=1}^{m}\sum_{j=1}^{l} B_{ij}s_i r_j - \lambda\sum_{i=1}^{m} d_i^{(X)} s_i^2 - \mu\sum_{j=1}^{l} d_j^{(Y)} r_j^2\right] = 0,$$

and $$\frac{\partial}{\partial r_{j'}}\left[\sum_{i=1}^{m}\sum_{j=1}^{l} B_{ij}s_i r_j - \lambda\sum_{i=1}^{m} d_i^{(X)} s_i^2 - \mu\sum_{j=1}^{l} d_j^{(Y)} r_j^2\right] = 0,$$

$$\Rightarrow \sum_{j=1}^{l} B_{ij} r_j - 2\lambda d_i^{(X)} s_i = 0, \tag{A6}$$

$$\text{and} \quad \sum_{i=1}^{m} B_{ij} s_i - 2\mu d_j^{(Y)} r_j = 0. \tag{A7}$$

Hence, taking $\mathbf{D}^{(X)}$ and $\mathbf{D}^{(Y)}$ as the diagonal matrices with the degree vectors $\mathbf{d}^{(X)}$ and $\mathbf{d}^{(Y)}$, respectively, on their leading diagonals,

$$\mathbf{B}\mathbf{r} = 2\lambda \mathbf{D}^{(x)}\mathbf{s} \tag{A8}$$

and

$$\mathbf{B}^\top \mathbf{s} = 2\mu \mathbf{D}^{(y)}\mathbf{r}. \tag{A9}$$

Substituting for $\mathbf{s}$, Eq. (A9) in Eq. (A8), gives

$$(\mathbf{D}^{(y)})^{-1}\mathbf{B}^\top(\mathbf{D}^{(x)})^{-1}\mathbf{B}\mathbf{r} = 4\lambda\mu\mathbf{r}, \tag{A10}$$

$$\Rightarrow (\mathbf{D}^{(y)})^{-1/2}\mathbf{B}^\top(\mathbf{D}^{(x)})^{-1/2}(\mathbf{D}^{(x)})^{-1/2}\mathbf{B}(\mathbf{D}^{(y)})^{-1/2}\mathbf{r} = 4\lambda\mu\mathbf{r},$$

$$\Rightarrow ((\mathbf{D}^{(x)})^{-1/2}\mathbf{B}(\mathbf{D}^{(y)})^{-1/2})^\top((\mathbf{D}^{(x)})^{-1/2}\mathbf{B}(\mathbf{D}^{(y)})^{-1/2})\mathbf{r} = 4\lambda\mu\mathbf{r}, \tag{A11}$$

$$\Rightarrow \mathbf{M}^\top\mathbf{M}\mathbf{r} = 4\lambda\mu\mathbf{r}, \tag{A12}$$

where

$$\mathbf{M} = (\mathbf{D}^{(x)})^{-1/2}\mathbf{B}(\mathbf{D}^{(y)})^{-1/2}.$$

By an identical argument, substituting Eq. (A8) in Eq. (A9) and rearranging equivalently,

$$\mathbf{M}\mathbf{M}^\top\mathbf{s} = 4\lambda\mu\mathbf{s}. \tag{A13}$$

Hence, $\mathbf{s}$ and $\mathbf{r}$ are eigenvectors of $\mathbf{M}\mathbf{M}^\top$ and $\mathbf{M}^\top\mathbf{M}$, respectively, with $4\lambda\mu$ the corresponding eigenvalue in both cases. Therefore, $\mathbf{s}$ and $\mathbf{r}$ are left and right singular vectors, respectively, of

$$\mathbf{M} = (\mathbf{D}^{(x)})^{-1/2}\mathbf{B}(\mathbf{D}^{(y)})^{-1/2},$$

with corresponding singular value $2\sqrt{\lambda\mu}$.

Multiplying Eq. (A6) by $s_i/2d^{++}$, summing over i, and referring to Eq. (A4) gives

$$\frac{1}{2d^{++}}\sum_{i=1}^{m}\sum_{j=1}^{l} B_{ij}s_i r_j = \frac{2\lambda}{2d^{++}}\sum_{i=1}^{m} d_i^{(X)} s_i^2 = \frac{2\lambda d^{++}}{2d^{++}} = \lambda.$$

Hence, referring to Eq. (A3), we get

$$Q_{XY} = \lambda. \tag{A14}$$

Then equivalently multiplying Eq. (A7) by $r_j/2d^{++}$, summing over j, and referring to Eq. (A5), and then referring to Eq. (A3) gives

$$Q_{XY} = \mu. \tag{A15}$$

Therefore, referring again to Eqs. (A12) and (A13), the maximum modularity solution is for the left and right singular vectors of M which correspond to the greatest singular value 2λ.

Now substituting Eq. (7) in Eq. (A9), we get

$$\mathbf{s}^\top\left(\mathbf{A} - \frac{1}{d^{++}}\mathbf{d}^{(X)}(\mathbf{d}^{(Y)})^\top\right) = 2\mu\mathbf{r}^\top\mathbf{D}^{(y)},$$

$$\Rightarrow \mathbf{s}^\top\mathbf{A} = \frac{1}{d^{++}}\mathbf{s}^\top\mathbf{d}^{(X)}(\mathbf{d}^{(Y)})^\top + 2\mu\mathbf{r}^\top\mathbf{D}^{(y)}. \tag{A16}$$

Post-multiplying Eq. (A16) by $\mathbf{1} = (1, 1, 1, \ldots)$ leads to

$$\mathbf{s}^\top\mathbf{d}^{(X)} = \frac{1}{d^{++}}\mathbf{s}^\top\mathbf{d}^{(X)}d^{++} + 2\mu\mathbf{r}^\top\mathbf{d}^{(Y)}$$

$$\therefore \quad \mu\mathbf{r}^\top\mathbf{d}^{(Y)} = 0.$$

Assuming that there is co-community structure present in $\mathbf{A}$, there must be positive comodularity, i.e., $Q_{XY} > 0 \Rightarrow \mu > 0$ [referring back to Eq. (A15)], and therefore $\mathbf{r}^\top\mathbf{d}^{(Y)} = 0$. By an identical argument, also $\mathbf{s}^\top\mathbf{d}^{(X)} = 0$. Therefore, for eigenvectors $\mathbf{r}$ corresponding to $Q_{XY} > 0$,

$$\mathbf{B}\mathbf{r} = \left(\mathbf{A} - \frac{1}{d^{++}}\mathbf{d}^{(X)}(\mathbf{d}^{(Y)})^\top\right)\mathbf{r} = \mathbf{A}\mathbf{r},$$

and so to find these eigenvectors with Q_{XY} maximized, instead of Eq. (A11) we can consider

$$((\mathbf{D}^{(x)})^{-1/2}\mathbf{A}(\mathbf{D}^{(y)})^{-1/2})^{\top}((\mathbf{D}^{(x)})^{-1/2}\mathbf{A}(\mathbf{D}^{(y)})^{-1/2})\mathbf{r} = (2\lambda)^2\mathbf{r} \tag{A17}$$

which, referring back to Eq. (6), can be written in terms of the co-Laplacian $\mathbf{L}_{XY}$ as

$$\mathbf{L}_{XY}^{\top}\mathbf{L}_{XY}\mathbf{r} = (2\lambda)^2\mathbf{r}.$$

By identical argument, we can also write

$$((\mathbf{D}^{(x)})^{-1/2}\mathbf{A}(\mathbf{D}^{(y)})^{-1/2})((\mathbf{D}^{(x)})^{-1/2}\mathbf{A}(\mathbf{D}^{(y)})^{-1/2})^{\top}\mathbf{s} = (2\lambda)^2\mathbf{s} \tag{A18}$$

and

$$\mathbf{L}_{XY}\mathbf{L}_{XY}^{\top}\mathbf{s} = (2\lambda)^2\mathbf{s}.$$

Hence, the co-Laplacian $\mathbf{L}_{XY}$ has left and right singular vectors $\mathbf{s}$ and $\mathbf{r}$, respectively, with corresponding singular values 2λ. It can be seen that Eq. (A17) has the eigenvector $\mathbf{1} = (1, 1, 1, \ldots)$, as follows:

$$\begin{aligned}
((\mathbf{D}^{(x)})^{-1/2}\mathbf{A}(\mathbf{D}^{(y)})^{-1/2})^{\top}((\mathbf{D}^{(x)})^{-1/2}\mathbf{A}(\mathbf{D}^{(y)})^{-1/2})\mathbf{1} &= (2\lambda)^2\mathbf{1} \\
\Rightarrow (\mathbf{D}^{(y)})^{-1}\mathbf{A}^{\top}(\mathbf{D}^{(x)})^{-1}\mathbf{A}\mathbf{1} &= (2\lambda)^2\mathbf{1} \\
\Rightarrow (\mathbf{D}^{(y)})^{-1}\mathbf{A}^{\top}(\mathbf{D}^{(x)})^{-1}\mathbf{d}^{(X)} &= (2\lambda)^2\mathbf{1} \\
\Rightarrow (\mathbf{D}^{(y)})^{-1}\mathbf{A}^{\top}\mathbf{1} &= (2\lambda)^2\mathbf{1} \\
\Rightarrow (\mathbf{D}^{(y)})^{-1}\mathbf{d}^{(Y)} &= (2\lambda)^2\mathbf{1} \\
\mathbf{1} &= (2\lambda)^2\mathbf{1}
\end{aligned}$$

and hence the corresponding eigenvalue is $(2\lambda)^2 = 1$, which, by the Perron-Frobenius theorem, must be the greatest eigenvalue [33,64]. An identical argument can also be applied to $\mathbf{s}$ in Eq. (A18). This means that the greatest singular value $2\lambda = 1$ corresponds to these left and right singular vectors which are both $\mathbf{1}$ (of lengths m and l, respectively); however, such singular vectors do not satisfy $\mathbf{r}^{\top}\mathbf{d}^{(Y)} = 0$ and $\mathbf{s}^{\top}\mathbf{d}^{(X)} = 0$. Therefore, to maximize the comodularity in the case of two co-communities, we should divide the X and Y nodes according to the left and right singular vectors, respectively, which correspond to the second greatest singular value.

The above explains how Algorithm 1 works for the case of two co-communities. An equivalent extension to k communities has been made in the unipartite community-detection setting [36]. To do so, the community labels are identified with the vertices of $k-1$ simplices; i.e., for detection of three communities, the co-community labels would be the vertices of a triangle. Relaxing constraints equivalent to Eqs. (A4) and (A5) means allowing the nodes to move away from the vertices of the simplex. This amounts to clustering the nodes in the space of the eigenvectors corresponding to the second to kth greatest eigenvalues of the Laplacian $\mathbf{L}$. This clustering is conventionally done using k means. The reader is referred to [36] for the detailed technical derivations relating to this. A similar extension can naturally be made in this co-community-detection setting. To detect $k^{(X)}$ X-node groupings, and $k^{(Y)}$ Y-node groupings, the X and Y nodes can be separately clustered (using k means independently for the X and Y nodes) in the spaces of the left and right singular vectors (respectively) corresponding to the second to $k^{(X)}$th and second to $k^{(Y)}$th greatest singular values, respectively, of the singular value decomposition of the co-Laplacian $\mathbf{L}_{XY}$.

APPENDIX B: PROOF OF PROPOSITION 1, FOR THE CASE OF TWO CO-COMMUNITIES

For the case of two co-communities, with θ_{in} and θ_{out} defined according to Eq. (2), with the co-community labels r_i and s_j defined as in Appendix A [Eqs. (A1) and (A2)], and with $G^{(X)}$ and $G^{(Y)}$ defined according to Definition 1, we note (equivalently to [33]) that

$$\theta_{z^{(X)}(i),z^{(Y)}(j)} = \tfrac{1}{2}(\theta_{\text{in}} + \theta_{\text{out}} + r_i s_j(\theta_{\text{in}} - \theta_{\text{out}})), \tag{B1}$$

and

$$\ln(\theta_{z^{(X)}(i),z^{(Y)}(j)}) = \frac{1}{2}\left(\ln(\theta_{\text{in}}\theta_{\text{out}}) + r_i s_j \ln\left(\frac{\theta_{\text{in}}}{\theta_{\text{out}}}\right)\right). \tag{B2}$$

We note that Eqs. (B1) and (B2) only hold because $s_i \in \{-1, 1\}$ and $r_j \in \{-1, 1\}$. Substituting Eqs. (B1) and (B2) into Eq. (3), and estimating the node-specific connectivity parameters $\boldsymbol{\pi}^{(X)}$ and $\boldsymbol{\pi}^{(Y)}$ by the degree distributions $\mathbf{d}^{(X)}$ and $\mathbf{d}^{(Y)}$, leads to the profile likelihood (using the Poisson approximation of [34])

$$\ell(\boldsymbol{\theta}; \mathbf{d}^{(X)}, \mathbf{d}^{(Y)}; G^{(X)}, G^{(Y)}) = \sum_{i=1}^{m}\sum_{j=1}^{l}\left[\frac{A_{ij}}{2}\left(\ln(\theta_{\text{in}}\theta_{\text{out}}) + r_i s_j \ln\left(\frac{\theta_{\text{in}}}{\theta_{\text{out}}}\right)\right) - \frac{d_i^{(X)}d_j^{(Y)}}{2}(\theta_{\text{in}} + \theta_{\text{out}} + r_i s_j(\theta_{\text{in}} - \theta_{\text{out}}))\right]$$

$$\begin{aligned}
\Rightarrow \ell(\boldsymbol{\theta}; \mathbf{d}^{(X)}, \mathbf{d}^{(Y)}; G^{(X)}, G^{(Y)}) = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{l}\Big[&A_{ij}\ln(\theta_{\text{in}}\theta_{\text{out}}) - d_i^{(X)}d_j^{(Y)}(\theta_{\text{in}} + \theta_{\text{out}}) \\
&+ \ln\left(\frac{\theta_{\text{in}}}{\theta_{\text{out}}}\right)\left(A_{ij} - d_i^{(X)}d_j^{(Y)}\cdot\frac{\theta_{\text{in}} - \theta_{\text{out}}}{\ln\theta_{\text{in}} - \ln\theta_{\text{out}}}\right)s_i r_j\Big].
\end{aligned}$$

We seek to maximize $\ell(\boldsymbol{\theta}; \mathbf{d}^{(X)}, \mathbf{d}^{(Y)}; G^{(X)}, G^{(Y)})$ with respect to $G^{(X)}$ and $G^{(Y)}$ by choosing the co-community labels s_i and r_j. Therefore, we can drop the terms constant in s_i and r_j to give

$$\widetilde{\ell}(\boldsymbol{\theta}; \mathbf{d}^{(X)}, \mathbf{d}^{(Y)}; G^{(X)}, G^{(Y)}) = \sum_{i=1}^{m}\sum_{j=1}^{l}\left(A_{ij} - d_i^{(X)}d_j^{(Y)}\cdot\frac{\theta_{\text{in}} - \theta_{\text{out}}}{\ln\theta_{\text{in}} - \ln\theta_{\text{out}}}\right)s_i r_j,$$

and defining

$$\eta = \frac{\theta_{\text{in}} - \theta_{\text{out}}}{\ln \theta_{\text{in}} - \ln \theta_{\text{out}}},$$

we therefore have

$$\widetilde{\ell}(\boldsymbol{\theta}; \mathbf{d}^{(X)}, \mathbf{d}^{(Y)}; G^{(X)}, G^{(Y)}) = \sum_{i=1}^{m}\sum_{j=1}^{l}\left(A_{ij} - \eta d_i^{(X)} d_j^{(Y)}\right) s_i r_j, \tag{B3}$$

which we note as equivalent to Eq. (22) in [33]. Proceeding similarly to that work, by applying to Eq. (B3) the constraints of Eqs. (A4) and (A5) with Lagrange multipliers λ and μ and differentiating and equating to zero, we get

$$\frac{\partial}{\partial s_{i'}}\left[\sum_{i=1}^{m}\sum_{j=1}^{l}\left(A_{ij} - \eta d_i^{(X)} d_j^{(Y)}\right) s_i r_j - \lambda \sum_{i=1}^{m} d_i^{(X)} s_i^2 - \mu \sum_{j=1}^{l} d_j^{(Y)} r_j^2\right] = 0,$$

$$\frac{\partial}{\partial r_{j'}}\left[\sum_{i=1}^{m}\sum_{j=1}^{l}\left(A_{ij} - \eta d_i^{(X)} d_j^{(Y)}\right) s_i r_j - \lambda \sum_{i=1}^{m} d_i^{(X)} s_i^2 - \mu \sum_{j=1}^{l} d_j^{(Y)} r_j^2\right] = 0,$$

$$\Rightarrow \sum_{j=1}^{l}\left(A_{ij} - \eta d_i^{(X)} d_j^{(Y)}\right) r_j - 2\lambda d_i^{(X)} s_i = 0, \quad \text{and} \quad \sum_{i=1}^{m}\left(A_{ij} - \eta d_i^{(X)} d_j^{(Y)}\right) s_i - 2\mu d_j^{(Y)} r_j = 0,$$

and therefore also recalling the definitions of $\mathbf{D}^{(X)}$ and $\mathbf{D}^{(Y)}$ as the diagonal matrices with the degree vectors $\mathbf{d}^{(X)}$ and $\mathbf{d}^{(Y)}$, respectively, on their leading diagonals,

$$(\mathbf{A} - \eta \mathbf{d}^{(X)} (\mathbf{d}^{(Y)})^{\top})\mathbf{r} = 2\lambda \mathbf{D}^{(X)} \mathbf{s} \tag{B4}$$

and

$$(\mathbf{A}^{\top} - \eta \mathbf{d}^{(Y)} (\mathbf{d}^{(X)})^{\top})\mathbf{s} = 2\mu \mathbf{D}^{(Y)} \mathbf{r}. \tag{B5}$$

Combining Eqs. (B4) and (B5) by substituting for $\mathbf{s}$ and $\mathbf{r}$, and following simplification identical to Eqs. (A10) and (A11), gives

$$\mathbf{W}^{\top}\mathbf{W}\mathbf{r} = 4\lambda\mu\mathbf{r}$$

and

$$\mathbf{W}\mathbf{W}^{\top}\mathbf{s} = 4\lambda\mu\mathbf{s},$$

where

$$\mathbf{W} = (\mathbf{D}^{(X)})^{-1/2}(\mathbf{A} - \eta \mathbf{d}^{(X)}(\mathbf{d}^{(Y)})^{\top})(\mathbf{D}^{(Y)})^{-1/2}.$$

Hence $\mathbf{s}$ and $\mathbf{r}$ are left and right singular vectors of the singular value decomposition of $\mathbf{W}$, again with corresponding singular values $4\lambda\mu$. Defining $\mathbf{1} = (1, 1, 1, \ldots)$, and noting that $\mathbf{1}\mathbf{d}^{(X)}(\mathbf{d}^{(Y)})^{\top} = d^{++}(\mathbf{d}^{(Y)})^{\top}$, etc., we can see that premultiplying Eqs. (B4) and (B5) by $\mathbf{1}$ leads to

$$\mathbf{r}^{\top}\mathbf{d}^{(Y)}(1 - d^{++}\eta) = 2\lambda \mathbf{s}^{\top}\mathbf{d}^{(X)} \tag{B6}$$

and

$$\mathbf{s}^{\top}\mathbf{d}^{(X)}(1 - d^{++}\eta) = 2\mu \mathbf{r}^{\top}\mathbf{d}^{(Y)}. \tag{B7}$$

Substituting for $\mathbf{s}^{\top}\mathbf{d}^{(X)}$ and $\mathbf{r}^{\top}\mathbf{d}^{(Y)}$, Eq. (B7) in Eq. (B6) and vice versa, gives

$$\mathbf{s}^{\top}\mathbf{d}^{(X)}[(1 - d^{++}\eta)^2 - 4\mu\lambda] = 0$$

and

$$\mathbf{r}^{\top}\mathbf{d}^{(Y)}[(1 - d^{++}\eta)^2 - 4\mu\lambda] = 0,$$

and therefore because $(1 - d^{++}\eta)^2 - 4\mu\lambda$ is not guaranteed to be zero,

$$\mathbf{s}^{\top}\mathbf{d}^{(X)} = 0$$

and

$$\text{and} \quad \mathbf{r}^{\top}\mathbf{d}^{(Y)} = 0.$$

Therefore, Eqs. (B4) and (B5) reduce to

$$\mathbf{A}\mathbf{r} = 2\lambda \mathbf{D}^{(x)}\mathbf{s}$$

and

$$\mathbf{A}^{\top}\mathbf{s} = 2\mu \mathbf{D}^{(y)}\mathbf{r},$$

and again combining these equations by substituting for $\mathbf{s}$ and $\mathbf{r}$ and following equivalent simplification to Eqs. (A10) and (A11), we hence find that $\mathbf{s}$ and $\mathbf{r}$ are left and right singular vectors of the co-Laplacian [Eq. (6)]. Therefore, the choice of the co-community labels $\mathbf{s}$ and $\mathbf{r}$ which maximizes the model likelihood specified in Eq. (3), subject also to the constraint of Eq. (2), is equivalent to the maximum comodularity assignment obtained via Algorithm 1.

APPENDIX C: SELECTING THE NUMBER OF CO-COMMUNITIES

In order to use Algorithm 1 to carry out co-community detection, we must specify the number of X-node groupings $k^{(X)}$, and the number of Y-node groupings $k^{(Y)}$.

If we want to let the network grow, it would be impractical to fully specify more complicated versions of the parametric model of Definition 1, which completely account for all effects. Instead, we can make a nonparametric generalization of this model incorporating more smoothing, based on the notion of the graphon. The graphon is a latent, smooth function which sets the probability between each pair of nodes of a connection forming between that pair of nodes [61]. In this

setting, the graphon is not symmetric, due to the two different types of nodes modeled.

Definition 7. For the Lipschitz-continuous graphon function $f \in L((0,1)^2)$, with $\mathbf{A}$ defined according to Definition 1, define connectivity functions $\phi^{(X)} \in L(0,1)$ and $\phi^{(Y)} \in L(0,1)$, and define latent orderings $\xi_i^{(X)} \overset{i.i.d.}{\sim} \mathcal{U}(0,1)$ and (independently) $\xi_j^{(Y)} \overset{i.i.d.}{\sim} \mathcal{U}(0,1)$ on the graphon margins of X and Y nodes $i \in \{1,\dots,m\}$ and $j \in \{1,\dots,l\}$, respectively. Then,

$$\mathbb{E}(A_{ij}) = f\big(\xi_i^{(X)}, \xi_j^{(Y)}\big)\phi^{(X)}\big(\xi_i^{(X)}\big)\phi^{(Y)}\big(\xi_j^{(Y)}\big). \tag{C1}$$

The graphon f (Definition 7) can be considered an infinite-dimensional equivalent to $\theta_{p,q}$ (Definition 1), up to a reordering of the nodes (after the Aldous-Hoover theorem [29]). The connectivity functions $\phi^{(X)}$ and $\phi^{(Y)}$ (Definition 7) are then similarly equivalent to the node-specific connectivity parameters $\boldsymbol{\pi}^{(X)}$ and $\boldsymbol{\pi}^{(Y)}$ (Definition 1 for the degree-corrected stochastic coblockmodel). These functions $\phi^{(X)}$ and $\phi^{(Y)}$ model the general variability of connectivity strength throughout the network, whereas the graphon f models the tendency for regions of the network to aggregate into specific co-communities. The model of Definition 7 is a more general model which is specified similarly for any network size. Thus, Eq. (C1) contains redundancy, and hence, as the networks we consider here are of fixed size, the degree-corrected stochastic coblockmodel (Definition 1) may be a more parsimonious choice. To estimate the generating mechanism of a bipartite network stably, Definition 7 must be replaced by a model with a limited number of parameters, i.e., Definition 1.

The network histogram method of fitting the stochastic blockmodel [16] in the unipartite or symmetric community-detection setting provides a rule-of-thumb method for selecting the optimal number of communities, or blocks, in the model. However, we note that this rule-of-thumb method may not result in a perfect match between the number of communities and the size of blocks, as spectral clustering may not result in equal size communities. Fitted in this way, the blockmodel is a valid representation of a network, whatever the generating mechanism of that network, as long as this generating mechanism results in an exchangeable network. The network histogram approximates the graphon, which is a continuous function: the nodes correspond to discrete locations along the graphon margins, ordered in an optimal way to satisfy the smoothness requirement of the graphon. The graphon oracle [16,61] defines a good ordering of the nodes, according to graphon smoothness, and coassociation patterns. This information is not available in practice, but it can be used to bound the mean integrated squared error of the network histogram approximation to the graphon. This ordering naturally corresponds to community assignments, and the number of communities, or blocks, is determined by the smoothness of the graphon. An intuition for this is by analogy with a wave: if there are many peaks over a fixed distance (i.e., short wavelength), the maximum gradient of the wave will be large, whereas if there are few peaks over the same fixed distance (i.e., long wavelength), the maximum gradient will be small. Similarly, the more communities, or peaks, that there are in the graphon, the greater the maximum gradient of the graphon will be and, correspondingly, the less smooth it will be.

1. Finding the optimal numbers of X- and Y-node groupings

In this section we define the anisotropic graphon, which allows us to determine an optimal number of X- and Y-node groupings, $k^{(X)}$ and $k^{(Y)}$, from which co-communities can be identified. This relates closely to the network histogram method in the symmetric unipartite community-detection setting [16]. In the unipartite community-detection setting, the graphon is a symmetric limit object bounded on $(0,1)^2$. It is symmetric because in that setting, the set of X nodes is the same as the set of Y nodes, and hence the smoothness is the same with respect to the corresponding orthogonal directions on the graphon. In contrast, in this co-community-detection setting the graphon is asymmetric, having different smoothnesses with respect to the X and Y nodes. Hence, we refer to this as the "anisotropic graphon," which is similarly a limit object bounded on $(0,1)^2$. To aid our analyses, we can stretch the anisotropic graphon so that it has the same smoothness with respect to the X nodes, and with respect to the Y nodes. It is easy to see that such a transformation exists for all anisotropic graphons. We refer to the result of stretching the anisotropic graphon in this way as the "equi-smooth graphon." Without loss of generality, this transformation can be expressed as a stretch of scale factor γ with respect to the X nodes, and a simultaneous stretch of scale factor $1/\gamma$ with respect to the Y nodes. We refer to γ as the anisotropy factor. This is formalized as follows.

Definition 8. For the Lipschitz-continuous anisotropic graphon $f \in L((0,1)^2)$ defined according to Definition 7, let the anisotropy factor γ define the linear-stretch transformation which maps f onto the Lipschitz-continuous equi-smooth graphon $\widetilde{f} \in L((0,\gamma) \times (0,1/\gamma))$. Then,

$$f(x,y) = \widetilde{f}(\gamma x, y/\gamma). \tag{C2}$$

Lipschitz continuity, in this context, means that the smoothness of the graphon (anisotropic or equi-smooth) is upper bounded, and we use this bound to calculate the optimal number of X- and Y-node groupings. We note that as the graphon is asymmetric in the bipartite case, this assignation is still suitable.

To determine the optimal number of X- and Y-node groupings, $k^{(X)}$ and $k^{(Y)}$, assuming a fixed block size to do the mapping, we set these $k^{(X)}$ and $k^{(Y)}$ so as to minimize the mean integrated squared error (MISE) of the blockmodel approximation of the graphon. Following a methodology which is closely related to the network histogram estimator in the symmetric (unipartite) community-detection setting [16], making use of the graphon oracle estimator, an upper bound can be calculated on this MISE, from a bias-variance decomposition, as follows:

Lemma 1. With $\mathbf{A}$, m, l, $g^{(X)} \in G^{(X)}$, and $g^{(Y)} \in G^{(Y)}$ defined according to Definition 1, let ρ be a deterministic scaling constant which specifies the expected number of edges in the network, such that

$$\rho = \mathbb{E}\left(\frac{1}{ml}\sum_{j=1}^{l}\sum_{i=1}^{m} A_{ij}\right),$$

and define piecewise block approximations to the adjacency matrix, for each pairing of a set of X nodes $g^{(X)}$ with a set of Y nodes $g^{(Y)}$, as

$$\bar{A}_{p,q} = \frac{\sum_{i \in g_p^{(X)}, j \in g_q^{(Y)}} A_{ij}}{\left|g_p^{(X)}\right| \left|g_q^{(Y)}\right|},$$

where $|\cdot|$ represents cardinality. With $z^{(X)}$ and $z^{(Y)}(j)$ defined according to Definition 1, $\xi^{(X)}$ and $\xi^{(Y)}$ defined according to Definition 7, and f defined according to Definition 8, define alternative map functions $\widetilde{z}^{(X)}(i')$, $i' \in \{1, \ldots, m\}$, and $\widetilde{z}^{(Y)}(j')$, $j' \in \{1, \ldots, l\}$. These map functions take the ordered locations of the X and Y nodes, respectively, along the graphon margins, as specified by $\xi^{(X)}$ and $\xi^{(Y)}$, and return the corresponding X- and Y-node groupings, such that $\widetilde{z}^{(X)}(\lceil \lceil]m \cdot \xi_i^{(X)}) = z^{(X)}(i)$, and $\widetilde{z}^{(Y)}(\lceil \lceil]l \cdot \xi_j^{(Y)}) = z^{(Y)}(j)$. Define the graphon oracle estimator as

$$\hat{f}(x, y) = \hat{\rho}^{-1} \bar{A}_{\widetilde{z}^{(X)}(\lceil lx \rceil), \widetilde{z}^{(Y)}(\lceil my \rceil)}, \tag{C3}$$

and let

$$\iint_{(0,1)^2} f(x, y) dx\, dy = 1. \tag{C4}$$

With $\widetilde{f}$ and γ defined as in Definition 8, let $\widetilde{M}$ be the maximum gradient of $\widetilde{f}$, and let $h^{(X)}$ and $h^{(Y)}$ be "bandwidth" parameters with respect to the X and Y nodes, respectively. Then, the graphon oracle upper bound on the MISE of the blockmodel estimate of the graphon function $\hat{f}$ is

$$\begin{aligned} \mathrm{MISE}(\hat{\mathrm{f}}) \leqslant \widetilde{M}^2 &\left\{ \gamma^2 \frac{(h^{(X)})^2}{m^2} + \frac{1}{\gamma^2} \frac{(h^{(Y)})^2}{l^2} \right\} \\ &+ 2\widetilde{M}^2 \left\{ \gamma^2 \frac{1}{4m} + \frac{1}{\gamma^2} \frac{1}{4l} \right\} \{1 + o(1)\} \\ &+ \frac{1}{\rho h^{(X)} h^{(Y)}} \{1 + o(1)\}. \end{aligned} \tag{C5}$$

Proof. See Appendix D. ∎

As would be expected from the form of the anisotropic graphon, this expression directly captures the resolution obtained in each axis. We note that the ordering of the nodes along the adjacency matrix margins is not necessarily the same as in their mappings along the graphon margins. Thus, we need to specify how nodes map to the groupings $g^{(X)}$ and $g^{(Y)}$ in a different way for the graphon, as compared to the adjacency matrix. This difference is accounted for by using different mapping functions: $\widetilde{z}^{(X)}(i')$ and $\widetilde{z}^{(Y)}(j')$ for the graphon, and $z^{(X)}(i)$ and $z^{(Y)}(j)$ for the adjacency matrix. That is, $\widetilde{z}^{(X)}(i')$ and $\widetilde{z}^{(Y)}(j')$ are required to specify the (contiguous) ranges and locations of the X- and Y-node groupings $g^{(X)}$ and $g^{(Y)}$ on the graphon margins, and equivalently $z^{(X)}(i)$ and $z^{(Y)}(j)$ for their (noncontiguous) locations on the adjacency matrix margins.

Using the MISE formulation of Lemma 1, we can estimate the optimal numbers of X- and Y-node groupings, $k^{(X)}$ and $k^{(Y)}$.

Proposition 2. With m and l defined as in Definition 1, and $\widetilde{M}$ and ρ defined as in Lemma 1, the optimal number of X- and Y-node groupings, $k^{(X)}$ and $k^{(Y)}$, respectively, are

$$k^{(X)} = \gamma (ml)^{\frac{1}{4}} (2\rho \widetilde{M}^2)^{\frac{1}{4}} \tag{C6}$$

and

$$k^{(Y)} = \frac{1}{\gamma} (ml)^{\frac{1}{4}} (2\rho \widetilde{M}^2)^{\frac{1}{4}}. \tag{C7}$$

Proof. The proof of this proposition is developed from the equivalent proof for the case of the isotropic graphon (corresponding to community detection in unipartite networks) [16]. The optimal bandwidths $h^{(X)*}$ and $h^{(Y)*}$ can be found by optimizing the expression for the MISE of Eq. (C5) with respect to $h^{(X)}$ and to $h^{(Y)}$ and setting to zero, and combining the resulting equations. To calculate $k^{(X)}$ and $k^{(Y)}$, substitute these optimal bandwidths $h^{(X)*}$ and $h^{(Y)*}$ into $k^{(X)} = m/h^{(X)*}$ and $k^{(Y)} = l/h^{(Y)*}$, which leads to Eqs. (C6) and (C7). ∎

We note that the above proof of Proposition 2 implies that X-node groupings are all the same size, and that the Y-node groupings are all the same size. This assumption is relaxed in the practical implementation of this methodology we propose: this point is discussed further in the next section.

2. Practical estimation of the number of X- and Y-node groupings

We implement spectral clustering by including a standard k-means step, to group the X and Y nodes in the spaces of the left and right singular vectors corresponding to the second to $k^{(X)}$th and second to $k^{(Y)}$th greatest singular values, respectively, of the singular value decomposition of the co-Laplacian $\mathbf{L}_{XY}$ [Eq. (6)]. This k-means step does not produce identical group sizes; however, we note that the estimates of $k^{(X)}$ and $k^{(Y)}$ defined according to Eqs. (C6) and (C7) assume that the X and Y node groupings are the same size (i.e., that the blocks in the blockmodel are all the same size with respect to the X nodes, and separately with respect to the Y nodes). We relax this requirement in practice (while noting that we must maintain $\min_i h_i^{(X)} / \max_i h_i^{(X)} = \Theta(1)$), because after examining several empirical data sets of the type presented in the next section, we observed that the group sizes produced by this type of regularized degree-corrected spectral clustering tend not to vary significantly in size (there are no "giant clusters"). Further, this requirement of identical group sizes is not physically realistic in the practical examples we present in the next section, and in many other real scenarios.

To estimate $\widetilde{M}$ and γ, we approximate the maximum slope of the graphon separately in the directions corresponding to the X and Y nodes, by considering the top component of the singular value decomposition of the adjacency matrix $\mathbf{A}$. This is equivalent to the rule-of-thumb procedure in the network histogram method, in the symmetric or unipartite community-detection scenario [16]. The top left and right singular vectors are ordered, and their gradients and values at their midpoints (the expected points of maximum slope) are estimated as $\hat{p}_X$ and $\hat{b}_X$, respectively, for the X nodes and $\hat{p}_Y$ and $\hat{b}_Y$, respectively, for the Y nodes. By thinking of this singular value decomposition as a factorization of the scaled, discretely sampled graphon (i.e., the ordered adjacency matrix), denoting the greatest singular value as ν leads to the linear approximations for the maximum gradient of the isotropic graphon M in the

directions of the X and Y nodes, M_X and M_Y, respectively:

$$\hat{M}_X = \frac{\nu}{\rho}\hat{p}_X\hat{b}_Y m, \quad \hat{M}_Y = \frac{\nu}{\rho}\hat{b}_X\hat{p}_Y l,$$

where m and l are the number of X and Y nodes, respectively (as previously defined). These factors m and l take account of the fact that the isotropic graphon margins are bounded on [0,1], whereas the adjacency matrix margins take the values $\{1,\dots,m\}$ and $\{1,\dots,l\}$, and the edge density factor ρ (defined as in Lemma 1) normalizes with respect to the adjacency matrix realization, such that the above estimates are independent of edge density ρ. The linear stretch transformation γ defines the maximum gradients of the equi-smooth graphon as $\widetilde{M}_X = \gamma M_X$ and $\widetilde{M}_Y = M_Y/\gamma$, respectively, and hence an estimate of the squared maximum gradient of the isotropic graphon can be found as

$$\hat{\widetilde{M}}^2 = \gamma^2\hat{M}_X{}^2 + \frac{1}{\gamma^2}\hat{M}_Y{}^2 = \frac{\nu^2}{\rho^2}\left(\gamma^2\hat{p}_X^2\hat{b}_Y^2 m^2 + \frac{1}{\gamma^2}\hat{b}_X^2\hat{p}_Y^2 l^2\right).$$

Using the assumption that the equi-smooth graphon is Lipschitz continuous, with the same upper bound on its smoothness with respect to both the X and Y nodes, i.e., $\widetilde{M}_X = \widetilde{M}_Y, \Rightarrow \gamma M_X = M_Y/\gamma$, we can estimate γ as

$$\hat{\gamma}^2 = \frac{\hat{M}_Y}{\hat{M}_X}. \tag{C8}$$

APPENDIX D: PROOF OF LEMMA 1

Define $\mathbf{A}$, $k^{(X)}$, and $k^{(Y)}$ according to Definition 1, define $\xi^{(X)}$ and $\xi^{(Y)}$ according to Definition 7, define f, $\widetilde{f}$, and γ according to Definition 8, and define ρ and $\widetilde{M}$ according to Lemma 1. Define bandwidths $h_p^{(X)} = |g_p^{(X)}|$ and $h_q^{(Y)} = |g_q^{(Y)}|$, where $|\cdot|$ represents cardinality, define $\omega(p,q)$ as the domain of integration over the block corresponding to the pairing of $g_p^{(X)}$ with $g_q^{(Y)}$ (where $g_p^{(X)}$ and $g_q^{(Y)}$ are sets of X nodes and Y nodes, with $G^{(X)}$ and $G^{(Y)}$ respectively the sets of $g_p^{(X)}$ and $g_q^{(Y)}$ over $p \in \{1,\dots,k^{(X)}\}$ and $q \in \{1,\dots,k^{(Y)}\}$), and define $\bar{A}_{p,q}$ as the block average corresponding to the pairing of $g_p^{(X)}$ with $g_q^{(Y)}$,

$$\bar{A}_{p,q} = \frac{\sum_{j\in g_q^{(Y)}}\sum_{i\in g_p^{(X)}} A_{ij}}{h_p^{(X)}h_q^{(Y)}}.$$

For convenience, we also define here theoretical relations to $\bar{A}_{p,q}$, by denoting the average values of f and f^2 over the block corresponding to the pairing of $g_p^{(X)}$ with $g_q^{(Y)}$ as $\bar{f}_{p,q}$ and $\bar{f^2}_{p,q}$, respectively:

$$\bar{f}_{p,q} = \frac{1}{|\omega(p,q)|}\iint_{\omega(p,q)} f(x,y)dx\,dy \tag{D1}$$

and

$$\bar{f^2}_{p,q} = \frac{1}{|\omega(p,q)|}\iint_{\omega(p,q)} f^2(x,y)dx\,dy, \tag{D2}$$

where

$$|\omega(p,q)| = \frac{h_p^{(X)}}{m}\frac{h_q^{(Y)}}{l}.$$

The bias-variance decomposition of the oracle MISE of the blockmodel approximation of the graphon function $\hat{f}$ can hence be written as [16]

$$\begin{aligned}\mathrm{MISE}(\hat{\mathrm{f}}) &\leqslant \mathbb{E}\iint_{(0,1)^2}\left|f(x,y)-\hat{f}(x,y)\right|^2 dx\,dy\\ &= \sum_{q=1}^{k^{(Y)}}\sum_{p=1}^{k^{(X)}}\iint_{\omega(p,q)}\left\{\left|f(x,y)-\frac{\mathbb{E}(\bar{A}_{p,q})}{\rho}\right|^2 + \frac{\mathrm{Var}(\bar{A}_{p,q})}{\rho^2}\right\}dx\,dy.\end{aligned} \tag{D3}$$

The domain of integration $\omega(p,q)$ is hence a contiguous region of the graphon, which corresponds to entries of the adjacency matrix which are not necessarily contiguous.

Modeling the equi-smooth graphon $\widetilde{f}$ as a linear stretch transformation of the anisotropic graphon f, by anisotropy factor γ, means that we can write

$$f(x,y) = \widetilde{f}(\gamma x, y/\gamma).$$

We define the graphon oracle [16,61] ordering of the X and Y nodes according to $\xi^{(X)}$ and $\xi^{(Y)}$, respectively. These are unobservable latent random vectors, which map the locations of the X and Y nodes from the margins of the graphon to the margins of the adjacency matrix. That is, $\xi_i^{(X)}$ and $\xi_j^{(Y)}$ provide the locations on the graphon margins which correspond to the X and Y nodes i and j, respectively, where i and j are the adjacency matrix indices of these nodes. We define $(i)^{-1}$ as a function which gives the rank of $\xi_i^{(X)}$, $1 \leqslant i \leqslant m$, and similarly $(j)^{-1}$ as a function which gives the rank of $\xi_j^{(Y)}$, $1 \leqslant j \leqslant l$. Therefore, $(i)^{-1}$ and $(j)^{-1}$ are functions which take the ordering along the adjacency matrix margins, and return the ordering along the graphon margins. Hence, the inverses of these functions, (i) and (j), take the ordering along the graphon margins, and return the corresponding ordering along the adjacency matrix margins. Adapting the proof of Lemma 3 from [16] to the anisotropic graphon, by defining $i_m = i/(m+1)$ and $j_l = j/(l+1)$, and assuming that $\widetilde{f}$ is Lipschitz continuous, gives

$$\left|f\left(\xi_{(i)}^{(X)},\xi_{(j)}^{(Y)}\right) - f(i_m,j_l)\right| = \left|\widetilde{f}\left(\gamma\xi_{(i)}^{(X)},\xi_{(j)}^{(Y)}/\gamma\right) - \widetilde{f}(\gamma i_m, j_l/\gamma)\right| \leqslant \widetilde{M}\left|\left(\gamma\xi_{(i)}^{(X)},\xi_{(j)}^{(Y)}/\gamma\right) - (\gamma i_m, j_l/\gamma)\right|.$$

Writing the variances and applying Jensen's inequality as in [16] we get

$$\mathrm{Var}\left(\xi_{(i)}^{(X)}\right) = \frac{i_m(1-i_m)}{m+2} \leqslant \frac{1/4}{m+2}, \quad \mathrm{Var}\left(\xi_{(j)}^{(Y)}\right) = \frac{j_l(1-j_l)}{l+2} \leqslant \frac{1/4}{l+2},$$

$$\Rightarrow \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left\{\gamma^2\left(\xi_{(i)}^{(X)}-i_m\right)^2 + \frac{1}{\gamma^2}\left(\xi_{(j)}^{(Y)}-j_l\right)^2\right\}^{\frac{1}{2}}$$

$$\leqslant \left(\gamma^2 \mathrm{Var}\left(\xi_{(i)}^{(X)}\right) + \frac{1}{\gamma^2}\mathrm{Var}\left(\xi_{(j)}^{(Y)}\right)\right)^{\frac{1}{2}} \leqslant \left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}},$$

$$\therefore \quad \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left|f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right) - f(i_m, j_l)\right| \leqslant \widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}. \tag{D4}$$

We note that this explains the stretching factor γ. Now adapting Lemma 2 from [16], we apply the law of iterated expectations to $A_{(i)(j)}$ to obtain

$$\mathbb{E}(A_{(i)(j)}) = \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\mathbb{E}_{A|\xi^{(X)},\xi^{(Y)}}\left(A_{(i)(j)}|\xi^{(X)}, \xi^{(Y)}\right)\right] = \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right]. \tag{D5}$$

Then using Jensen's inequality we get

$$\left|\mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right] - \rho f(i_m, j_l)\right| \leqslant \rho\mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\left|f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right) - f(i_m, j_l)\right|\right], \tag{D6}$$

and hence combining Eqs. (D4)–(D6), we have

$$\left|\mathbb{E}(A_{(i)(j)}) - \rho f(i_m, j_l)\right| \leqslant \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}. \tag{D7}$$

Now applying the law of total variance to $A_{(i)(j)}$, as in Lemma 2 from [16], we obtain

$$\begin{aligned}
\mathrm{Var}(A_{(i)(j)}) &= \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}[\mathrm{Var}_{A|\xi^{(X)},\xi^{(Y)}}(A_{(i)(j)}|\xi^{(X)}, \xi^{(Y)})] + \mathrm{Var}_{\xi^{(X)},\xi^{(Y)}}[\mathbb{E}_{A|\xi^{(X)},\xi^{(Y)}}(A_{(i)(j)}|\xi^{(X)}, \xi^{(Y)})] \\
&= \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\left(1 - \rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right)\right] + \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho^2\left(f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right)^2\right] - \left(\mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right]\right)^2 \\
&= \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right] - \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho^2\left(f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right)^2\right] + \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho^2\left(f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right)^2\right] \\
&\quad - \left(\mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right]\right)^2 \\
&= \mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right]\left\{\mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[1 - \rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right]\right\}.
\end{aligned} \tag{D8}$$

From Eq. (D4), we get

$$\mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right] \leqslant \rho f(i_m, j_l) + \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}} \tag{D9}$$

and

$$-\mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[\rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right] \leqslant -\rho f(i_m, j_l) + \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}, \tag{D10}$$

and hence also

$$\mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[1 - \rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right] \geqslant 1 - \rho f(i_m, j_l) - \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}} \tag{D11}$$

and

$$-\mathbb{E}_{\xi^{(X)},\xi^{(Y)}}\left[1 - \rho f\left(\xi_{(i)}^{(X)}, \xi_{(j)}^{(Y)}\right)\right] \geqslant -1 + \rho f(i_m, j_l) - \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}. \tag{D12}$$

Now combining Eq. (D9) with the negative of Eq. (D12) and applying Eq. (D8) we get

$$\mathrm{Var}(A_{(i)(j)}) \leqslant \left[\rho f(i_m, j_l) + \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\right]\left[1 - \rho f(i_m, j_l) + \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\right]$$

and hence

$$\mathrm{Var}(A_{(i)(j)}) \leqslant \rho f(i_m, j_l)[1 - \rho f(i_m, j_l)] + \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\left[1 + \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\right]. \tag{D13}$$

Similarly combining the negative of Eq. (D10) with Eq. (D11) and applying Eq. (D8) we get

$$\mathrm{Var}(A_{(i)(j)}) \geqslant \left[\rho f(i_m, j_l) - \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\right]\left[1 - \rho f(i_m, j_l) - \rho\widetilde{M}\left\{\gamma^2 \frac{1}{4(m+2)} + \frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\right],$$

and hence

$$\mathrm{Var}(A_{(i)(j)}) \geqslant \rho f(i_m, j_l)[1-\rho f(i_m, j_l)] - \rho\widetilde{M}\left\{\gamma^2\frac{1}{4(m+2)}+\frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\left[1-\rho\widetilde{M}\left\{\gamma^2\frac{1}{4(m+2)}+\frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\right],$$

and therefore

$$\begin{aligned}-\mathrm{Var}(A_{(i)(j)}) &\leqslant -\rho f(i_m, j_l)[1-\rho f(i_m, j_l)] + \rho\widetilde{M}\left\{\gamma^2\frac{1}{4(m+2)}+\frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\left[1-\rho\widetilde{M}\left\{\gamma^2\frac{1}{4(m+2)}+\frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\right]\\ &\leqslant -\rho f(i_m, j_l)[1-\rho f(i_m, j_l)] + \rho\widetilde{M}\left\{\gamma^2\frac{1}{4(m+2)}+\frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\left[1+\rho\widetilde{M}\left\{\gamma^2\frac{1}{4(m+2)}+\frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\right],\end{aligned} \tag{D14}$$

and hence combining Eqs. (D13) and (D14) we get

$$|\mathrm{Var}(A_{(i)(j)}) - \rho f(i_m, j_l)[1-\rho f(i_m, j_l)]| \leqslant \rho\widetilde{M}\left\{\gamma^2\frac{1}{4(m+2)}+\frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\left[1+\rho\widetilde{M}\left\{\gamma^2\frac{1}{4(m+2)}+\frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}^{\frac{1}{2}}\right]. \tag{D15}$$

Now referring to Eq. (D7) and comparing it to Eq. (6) of the Supporting Information Sec. A in [16] allows us to rewrite the covariance expression in Lemma 2 of [16], giving

$$\mathrm{Cov}(A_{(i)(j)}, A_{(i')(j')}) \leqslant \rho^2\widetilde{M}^2\left\{\gamma^2\frac{1}{4(m+2)}+\frac{1}{\gamma^2}\frac{1}{4(l+2)}\right\}, \tag{D16}$$

$i \neq i'$, $j \neq j'$. We can then use Eqs. (D7), (D15), and (D16) to adapt Proposition 1 from [16], also referring to Eqs. (D1) and (D2), to give

$$|\mathbb{E}(\bar{A}_{p,q}) - \rho\bar{f}_{p,q}| \leqslant \rho\widetilde{M}\left\{\gamma^2\frac{1}{4m}+\frac{1}{\gamma^2}\frac{1}{4l}\right\}^{\frac{1}{2}}\{1+o(1)\} \tag{D17}$$

and

$$\left|\mathrm{Var}(\bar{A}_{p,q}) - \frac{\rho\bar{f}_{p,q}-\rho^2\bar{f}^2{}_{p,q}}{h_p^{(X)}h_q^{(Y)}}\right| \leqslant \frac{\rho\widetilde{M}}{h_p^{(X)}h_q^{(Y)}}\left\{\gamma^2\frac{1}{4m}+\frac{1}{\gamma^2}\frac{1}{4l}\right\}^{\frac{1}{2}}\{1+o(1)\} + \rho^2\widetilde{M}^2\left\{\gamma^2\frac{1}{4m}+\frac{1}{\gamma^2}\frac{1}{4l}\right\}, \tag{D18}$$

which is a conservative upper bound. Now substituting Eq. (D18) back into Eq. (D3), we get

$$\begin{aligned}\mathrm{MISE}(\hat{\mathrm{f}}) \leqslant \sum_{q=1}^{k^{(Y)}}\sum_{p=1}^{k^{(X)}}\iint_{\omega(p,q)}\Bigg[&|\{f(x,y)-\bar{f}_{p,q}\}+\{\bar{f}_{p,q}-\mathbb{E}(\bar{A}_{p,q})/\rho\}|^2 + \frac{\bar{f}_{p,q}-\rho\bar{f}^2{}_{p,q}}{\rho h_p^{(X)}h_q^{(Y)}}\\ &+ \frac{\widetilde{M}}{\rho h_p^{(X)}h_q^{(Y)}}\left\{\gamma^2\frac{1}{4m}+\frac{1}{\gamma^2}\frac{1}{4l}\right\}^{\frac{1}{2}}\{1+o(1)\} + \widetilde{M}^2\left\{\gamma^2\frac{1}{4m}+\frac{1}{\gamma^2}\frac{1}{4l}\right\}\Bigg]dx\,dy,\end{aligned}$$

and then substituting Eq. (D17), integrating and rearranging, leads to

$$\begin{aligned}\mathrm{MISE}(\hat{\mathrm{f}}) \leqslant \sum_{q=1}^{k^{(Y)}}\sum_{p=1}^{k^{(X)}}\Bigg[&\iint_{\omega(p,q)}|f(x,y)-\bar{f}_{p,q}|^2dx\,dy + \Bigg(2\widetilde{M}^2\left\{\gamma^2\frac{1}{4m}+\frac{1}{\gamma^2}\frac{1}{4l}\right\}\{1+o(1)\} + \frac{\bar{f}_{p,q}-\rho\bar{f}^2{}_{p,q}}{\rho h_p^{(X)}h_q^{(Y)}}\\ &+ \frac{\widetilde{M}}{\rho h_p^{(X)}h_q^{(Y)}}\left\{\gamma^2\frac{1}{4m}+\frac{1}{\gamma^2}\frac{1}{4l}\right\}^{\frac{1}{2}}\{1+o(1)\}\Bigg)\frac{h_p^{(X)}}{m}\frac{h_q^{(Y)}}{l}\Bigg].\end{aligned} \tag{D19}$$

Then, adapting the proof of Lemma 1 from [16], we can write

$$|\bar{f}_{p,q} - f(x,y)| = \left|\frac{1}{|\omega(p,q)|}\iint_{\omega(p,q)}f(x',y')dx'\,dy' - f(x,y)\right| \leqslant \frac{1}{|\omega(p,q)|}\iint_{\omega(p,q)}\left|\widetilde{f}(\gamma x', y'/\gamma) - \widetilde{f}(\gamma x, y/\gamma)\right|dx'\,dy'.$$

Assuming $\widetilde{f}$ is Lipschitz continuous, it therefore follows that

$$|\bar{f}_{p,q} - f(x,y)| \leqslant \frac{1}{|\omega(p,q)|}\iint_{\omega(p,q)} \widetilde{M}|(\gamma x', y'/\gamma) - (\gamma x, y/\gamma)|dx'\,dy' \leqslant \frac{1}{|\omega(p,q)|}\iint_{\omega(p,q)} \widetilde{M}\sqrt{\gamma^2\frac{\left(h_p^{(X)}\right)^2}{m^2} + \frac{1}{\gamma^2}\frac{\left(h_q^{(Y)}\right)^2}{l^2}}dx'\,dy'$$

$$\Rightarrow |\bar{f}_{p,q} - f(x,y)| \leqslant \widetilde{M}\sqrt{\gamma^2\frac{\left(h_p^{(X)}\right)^2}{m^2} + \frac{1}{\gamma^2}\frac{\left(h_q^{(Y)}\right)^2}{l^2}}$$

and therefore

$$\frac{1}{|\omega(p,q)|}\iint_{\omega(p,q)} |\bar{f}_{p,q} - f(x,y)|^2 \leqslant \widetilde{M}^2\left\{\gamma^2\frac{\left(h_p^{(X)}\right)^2}{m^2} + \frac{1}{\gamma^2}\frac{\left(h_q^{(Y)}\right)^2}{l^2}\right\},$$

and hence summing over all the blocks corresponding to all pairings of X-node groupings $g^{(X)} \in G^{(X)}$ with Y-node groupings $g^{(Y)} \in G^{(Y)}$, and assuming $h^{(X)}$ and $h^{(Y)}$ are both constants, we get

$$\sum_{q=1}^{k^{(Y)}}\sum_{p=1}^{k^{(X)}}\iint_{\omega(p,q)} |\bar{f}_{p,q} - f(x,y)|^2 \leqslant \widetilde{M}^2\left\{\gamma^2\frac{(h^{(X)})^2}{m^2} + \frac{1}{\gamma^2}\frac{(h^{(Y)})^2}{l^2}\right\}. \tag{D20}$$

Recalling Eq. (D1) and Eq. (C4), i.e.,

$$\iint_{(0,1)^2} f(x,y)dx\,dy = 1,$$

and noting that

$$\sum_{q=1}^{k^{(Y)}}\sum_{p=1}^{k^{(X)}}\frac{\bar{f}_{p,q} - \rho\bar{f^2}_{p,q}}{\rho h_p^{(X)} h_q^{(Y)}} \leqslant \sum_{q=1}^{k^{(Y)}}\sum_{p=1}^{k^{(X)}}\frac{\bar{f}_{p,q}}{\rho h^{(X)} h^{(Y)}},$$

we can see that

$$\begin{aligned}\sum_{q=1}^{k^{(Y)}}\sum_{p=1}^{k^{(X)}}\frac{\bar{f}_{p,q} - \rho\bar{f^2}_{p,q}}{\rho h_p^{(X)} h_q^{(Y)}} &\leqslant \sum_{q=1}^{k^{(Y)}}\sum_{p=1}^{k^{(X)}}\frac{ml}{\rho(h^{(X)})^2(h^{(Y)})^2}\frac{h^{(X)}}{m}\frac{h^{(Y)}}{l}\bar{f}_{p,q}\\ &= \frac{ml}{\rho(h^{(X)})^2(h^{(Y)})^2}\sum_{q=1}^{k^{(Y)}}\sum_{p=1}^{k^{(X)}}\iint_{\omega(p,q)} f(x,y)dx\,dy\\ &= \frac{ml}{\rho(h^{(X)})^2(h^{(Y)})^2}\iint_{(0,1)^2} f(x,y)dx\,dy\\ &= \frac{ml}{\rho(h^{(X)})^2(h^{(Y)})^2}.\end{aligned} \tag{D21}$$

Now substituting Eqs. (D20) and (D21) into Eq. (D19), and rearranging, we get

$$\begin{aligned}\mathrm{MISE}(\hat{\mathrm{f}}) \leqslant{}& \widetilde{M}^2\left\{\gamma^2\frac{(h^{(X)})^2}{m^2} + \frac{1}{\gamma^2}\frac{(h^{(Y)})^2}{l^2}\right\} + 2\widetilde{M}^2\left\{\gamma^2\frac{1}{4m} + \frac{1}{\gamma^2}\frac{1}{4l}\right\}\{1+o(1)\}\\ &+ \frac{1}{\rho h^{(X)}h^{(Y)}} + \frac{\widetilde{M}}{\rho h^{(X)}h^{(Y)}}\left\{\gamma^2\frac{1}{4m} + \frac{1}{\gamma^2}\frac{1}{4l}\right\}^{\frac{1}{2}}\{1+o(1)\}\end{aligned}$$

and hence

$$\mathrm{MISE}(\hat{\mathrm{f}}) \leqslant \widetilde{M}^2\left\{\gamma^2\frac{(h^{(X)})^2}{m^2} + \frac{1}{\gamma^2}\frac{(h^{(Y)})^2}{l^2}\right\} + 2\widetilde{M}^2\left\{\gamma^2\frac{1}{4m} + \frac{1}{\gamma^2}\frac{1}{4l}\right\}\{1+o(1)\} + \frac{1}{\rho h^{(X)}h^{(Y)}}\{1+o(1)\}.$$

[1] P. W. Holland, K. B. Laskey, and S. Leinhardt, Stochastic blockmodels: First steps, Soc. Networks **5**, 109 (1983).

[2] P. J. Bickel and A. Chen, A nonparametric view of network models and Newman-Girvan and other modularities, Proc. Natl. Acad. Sci. USA **106**, 21068 (2009).

[3] K. Rohe, S. Chatterjee, and Bin Yu, Spectral clustering and the high-dimensional stochastic blockmodel, Ann. Stat. **39**, 1878 (2011).

[4] T. Qin and K. Rohe, Regularized spectral clustering under the degree-corrected stochastic blockmodel, in *Advances in Neural*

Information Processing Systems (NeurIPS, Lake Tahoe, 2013), pp. 3120–3128.

[5] J. D. Wilson, S. Wang, P. J. Mucha, S. Bhamidi, and A. B. Nobel, A testing based extraction algorithm for identifying significant communities in networks, Ann. Appl. Stat. **8**, 1853 (2014).

[6] Y. Zhang, E. Levina, and J. Zhu, Estimating network edge probabilities by neighborhood smoothing, arXiv:1509.08588.

[7] A. A. Amini, A. Chen, P. J. Bickel, and E. Levina, Pseudo-likelihood methods for community detection in large sparse networks, Ann. Stat. **41**, 2097 (2013).

[8] T. T. Cai and X. Li, Robust and computationally feasible community detection in the presence of arbitrary outlier nodes, Ann. Stat. **43**, 1027 (2015).

[9] M. E. Newman, Equivalence between modularity optimization and maximum likelihood methods for community detection, Phys. Rev. E **94**, 052315 (2016).

[10] L. Zhang and T. P. Peixoto, Statistical inference of assortative community structures, Phys. Rev. Research **2**, 043271 (2020).

[11] Y. Zhao, E. Levina, and J. Zhu, Consistency of community detection in networks under degree-corrected stochastic block models, Ann. Stat. **40**, 2266 (2012).

[12] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing, Mixed membership stochastic blockmodels, in *Advances in Neural Information Processing Systems* (NeurIPS, Vancouver, 2009), pp. 33–40.

[13] P. K. Gopalan and D. M. Blei, Efficient discovery of overlapping communities in massive networks, Proc. Natl. Acad. Sci. USA **110**, 14534 (2013).

[14] M. Girvan and M. E. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. USA **99**, 7821 (2002).

[15] P. Diaconis, Finite forms of de Finetti's theorem on exchangeability, Synthese **36**, 271 (1977).

[16] S. C. Olhede and P. J. Wolfe, Network histograms and universality of blockmodel approximation, Proc. Natl. Acad. Sci. USA **111**, 14722 (2014).

[17] M. A. Riolo, G. T. Cantwell, G. Reinert, and M. E. Newman, Efficient method for estimating the number of communities in a network, Phys. Rev. E **96**, 032310 (2017).

[18] T. P. Peixoto, Bayesian stochastic blockmodeling, *Advances in Network Clustering and Blockmodeling* (Wiley, Hoboken, 2019), pp. 289–332.

[19] T.-C. Yen and D. B. Larremore, Community detection in bipartite networks with stochastic block models, Phys. Rev. E **102**, 032309 (2020).

[20] P. Bickel, P. Diggle, S. Fienberg, U. Gather, I. Olkin, and S. Zeger, *Springer Series in Statistics* (Springer, Berlin, 2009).

[21] R. Tibshirani, G. Walther, and T. Hastie, Estimating the number of clusters in a data set via the gap statistic, J. R. Stat. Soc. Ser. B **63**, 411 (2001).

[22] C. J. Flynn and P. O. Perry, Consistent biclustering, Electron. J. Statist. **14**, 731 (2020).

[23] D. Choi and P. J. Wolfe, Co-clustering separately exchangeable network data, Ann. Stat. **42**, 29 (2014).

[24] S. C. Madeira and A. L. Oliveira, Biclustering algorithms for biological data analysis: A survey, IEEE/ACM Trans. Comput. Biol. Bioinf. **1**, 24 (2004).

[25] C. Gao, Y. Lu, Z. Ma, and H. H. Zhou, Optimal estimation and completion of matrices with biclustering structures, J. Mach. Learn. Res. **17**, 1 (2016).

[26] A. Bhattacharya and Y. Cui, A GPU-accelerated algorithm for biclustering analysis and detection of condition-dependent co-expression network modules, Sci. Rep. **7**, 4162 (2017).

[27] D.-A. Clevert, T. Unterthiner, G. Povysil, and S. Hochreiter, Rectified factor networks for biclustering of omics data, Bioinformatics **33**, i59 (2017).

[28] D. Rugeles, K. Zhao, C. Gao, M. Dash, and S. Krishnaswamy, Biclustering: An application of dual topic models, in *Proceedings of the 2017 SIAM International Conference on Data Mining* (Society for Industrial and Applied Mathematics, Philadelphia, 2017), pp. 453–461.

[29] D. J. Aldous, *Exchangeability and Related Topics* (Springer, Berlin, 1985).

[30] K. Rohe and B. Yu, Co-clustering for directed graphs: The stochastic co-blockmodel and a spectral algorithm, arXiv:1204.2296.

[31] M. E. Newman and M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E **69**, 026113 (2004).

[32] I. S. Dhillon, Co-clustering documents and words using bipartite spectral graph partitioning, in *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, San Francisco, 2001), pp. 269–274.

[33] M. E. Newman, Spectral methods for community detection and graph partitioning, Phys. Rev. E **88**, 042822 (2013).

[34] P. O. Perry and P. J. Wolfe, Null models for network data, arXiv:1201.5871.

[35] D. S. Bassett, M. A. Porter, N. F. Wymbs, S. T. Grafton, J. M. Carlson, and P. J. Mucha, Robust detection of dynamic community structure in networks, Chaos **23**, 013142 (2013).

[36] M. A. Riolo and M. Newman, First-principles multiway spectral partitioning of graphs, J. Complex Networks **2**, 121 (2014).

[37] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, Fast unfolding of communities in large networks, J. Stat. Mech. (2008) P10008.

[38] L. Page, S. Brin, R. Motwani, and T. Winograd, *The Pagerank Citation Ranking: Bringing Order to the Web* (Stanford InfoLab, 1999).

[39] D. C. Sørensen, Implicit application of polynomial filters in a k-step Arnoldi method, SIAM J. Matrix Anal. Appl. **13**, 357 (1992).

[40] R. B. Lehoucq and D. C. Sørensen, Deflation techniques for an implicitly restarted Arnoldi iteration, SIAM J. Matrix Anal. Appl. **17**, 789 (1996).

[41] A.-L. Barabási and Z. N. Oltvai, Network biology: Understanding the cell's functional organization, Nat. Rev. Genet. **5**, 101 (2004).

[42] A. Wagner, Estimating coarse gene network structure from large-scale gene perturbation data, Genome Res. **12**, 309 (2002).

[43] A.-L. Barabási and R. Albert, Emergence of scaling in random networks, Science **286**, 509 (1999).

[44] C. Borgs, J. T. Chayes, H. Cohn, and S. Ganguly, Consistent nonparametric estimation for heavy-tailed sparse graphs, Ann. Statistics **49**, 1904 (2021).

[45] S. C. Olhede and P. J. Wolfe, Degree-based network models, arXiv:1211.6537.
[46] O. Kallenberg, *Probabilistic Symmetries and Invariance Principles* (Springer, Berlin, 2005), Vol. 9.
[47] B. Bollobás, S. Janson, and O. Riordan, The phase transition in inhomogeneous random graphs, Random Struct. Alg. **31**, 3 (2007).
[48] B. Söderberg, General formalism for inhomogeneous random graphs, Phys. Rev. E **66**, 066121 (2002).
[49] Y. Benjamini and Y. Hochberg, Controlling the false discovery rate: A practical and powerful approach to multiple testing, J. R. Stat. Soc. Ser. B (Methodol.) **57**, 289 (1995).
[50] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, Comparing community structure identification, J. Stat. Mech. (2005) P09008.
[51] D. B. Larremore, A. Clauset, and A. Z. Jacobs, Efficiently inferring community structure in bipartite networks, Phys. Rev. E **90**, 012805 (2014).
[52] P. Jones, Functions of DNA methylation: Islands, start sites, gene bodies and beyond, Nat. Rev. Genet. **13**, 484 (2012).
[53] T. E. Bartlett, A. Zaikin, S. C. Olhede, J. West, A. E. Teschendorff, and M. Widschwendter, Corruption of the intragene DNA methylation architecture is a hallmark of cancer, PLoS One **8**, e68285 (2013).
[54] T. E. Bartlett, A. Jones, E. L. Goode, B. L. Fridley, J. M. Cunningham, E. M. Berns, E. Wik, H. B. Salvesen, B. Davidson, C. G. Trope *et al.*, Intra-gene DNA methylation variability is a clinically independent prognostic marker in women's cancers, PLoS One **10**, e0143178 (2015).
[55] T. E. Bartlett, Network inference and community detection, based on covariance matrices, correlations and test statistics from arbitrary distributions, Commun. Stat. Theory Methods **46**, 9150 (2017).
[56] T. Hampton, Cancer genome atlas, JAMA, J. Am. Med. Assoc. **296**, 1958 (2006).
[57] J. Leskovec, D. Huttenlocher, and J. Kleinberg, Signed networks in social media, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Association for Computing Machinery, Atlanta, 2010), pp. 1361–1370.
[58] Data source, http://www.broadinstitute.org/gsea/msigdb/
[59] A. Subramanian, P. Tamayo, V. Mootha, S. Mukherjee, B. Ebert, M. Gillette, A. Paulovich, S. Pomeroy, T. Golub, E. Lander *et al.*, Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles, Proc. Natl. Acad. Sci. USA **102**, 15545 (2005).
[60] Data source, http://grouplens.org/datasets/movielens/
[61] P. J. Wolfe and S. C. Olhede, Nonparametric graphon estimation, arXiv:1309.5936.
[62] P. Latouche, E. Birmelé, and C. Ambroise, Overlapping stochastic block models with application to the French political blogosphere, Ann. Appl. Stat. **5**, 309 (2011).
[63] H. Zanghi *et al.*, Strategies for online inference of model-based clustering in large and growing networks, Ann. Appl. Stat. **4**, 687 (2010).
[64] R. A. Hom and C. R. Johnson, *Topics in Matrix Analysis* (Cambridge University Press, New York, 1991).