Community detection with the canonical ensemble

Rudy Arthur*
Department of Computer Science, University of Exeter, Exeter EX4 4PY, United Kingdom

(Received 8 May 2026; accepted 18 August 2026; published 2 September 2026)

Network community detection is usually considered as an unsupervised learning problem. Given a network, the aim is to partition it using some general-purpose algorithm. In this paper, we instead treat community detection as a hypothesis testing problem. Given a network, we examine the evidence for community structure in the observed network compared to a network generated by a null model. To do this, we define an appropriate test statistic, analogous to a z-score, and discuss several null models derived from maximizing entropy under different constraints. These constraints are imposed in expectation, giving a canonical ensemble rather than the usual microcanonical ensemble. We demonstrate the application of this method on real and synthetic data and contrast our method to Bayesian approaches based on the stochastic block model. We demonstrate that this method gives rigorous answers to concrete questions, which can be more useful to analysts than the output of a generic algorithm.

DOI: 10.1103/rt96-mb8q

I. INTRODUCTION

Community detection involves partitioning the nodes of a network into groups. Generally, communities are defined such that nodes are densely connected within a community and sparsely connected between them [1]. Communities are the most important of the various mesoscale structures which can be used to summarize network data, though core-periphery [2], disassortative [3], and other patterns are also studied [4–6].

Community detection in networks shares many similarities with clustering vector data, with numerous methods for transforming one problem to the other [7,8]. Community detection and clustering also share a number of philosophical issues regarding the interpretation of the output of a community detection/clustering algorithm [9]. In this paper, we approach community detection from the point of view of [10] that clustering should not be treated as an application-independent mathematical problem, but should always be studied in the context of its end-use.

This means, rather than presuming there is a single correct partition of a network which some universal algorithm can discover, community detection should provide a way to probe a network with questions motivated by the origin and intended application of the data. We do this in the framework of statistical hypothesis testing, which is well established in community detection [11–13]. Hypothesis testing compares the value of some test statistic to the distribution of that statistic under the null. If the observed value is very unlikely to be generated by the null (quantified by the p-value), then we can reject the null and claim that the statistic is significant, i.e., the network has some structure not explained by the null.

Newman's modularity function [14], or some variant of it, is often used as the test statistic [11,15–17]. However, modularity uses a null model which is often invalid; it only detects assortative community structures and is sensitive to absolute rather than relative differences from the null expectation. We will discuss these issues in the next section, and the first contribution of this work is a test statistic that can be used instead of modularity, correcting these issues. We can use this statistic to detect communities or any other block structure, i.e., patterns like assortative/disassortative/core-periphery and others can also be detected. This formulation also allows the possibility of unassigned nodes, something common in vector clustering approaches [18–20] but rare in network clustering algorithms.

The second contribution of this paper is a thorough consideration of null models appropriate for community detection. We derive null models by maximizing entropy under constraints. Crucially, we use the canonical ensemble, where constraints are imposed on average, rather than the microcanonical ensemble, where constraints are imposed exactly. These are not equivalent choices [21]. The microcanonical ensemble is typically used in community detection, but the canonical ensemble is the appropriate one for networks which may have any uncertainty in their measurement.

We then provide a scheme for formal hypothesis testing of communities under a null model and demonstrate that our framework achieves theoretical bounds on synthetic data. We contrast our method on real data against alternative Bayesian methods which are strongly advocated by some [22]. We show that in the cases studied, we obtain more useful and interpretable results from our approach.

To be clear, this paper is not simply proposing new functions to maximize or a way to fix modularity. While we introduce a new quality function and a variety of null models, a rigorous approach to community detection requires more

*Contact author: R.Arthur@exeter.ac.uk

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

than simply maximizing a quality function. The primary goal of this paper is to introduce a practical scheme to ask and answer questions about the mesoscale structure of networks. These questions should be motivated by the origin and intended application of the data. The aim is to advocate a more thoughtful approach to community detection in applications, rather than current practice, which often involves throwing the network into the author's preferred black-box algorithm, then performing a *post hoc* analysis of the returned partition.

In Sec. II, we define our test statistic. In Sec. III, we derive our null models through entropy maximisation with constraints. Section IV describes the community detection procedure and implementation details. In Sec. V, we study different artificial examples, showing that this method agrees with theoretical expectations for community detectability. We compare our approach to the method of stochastic block model (SBM) inference in Sec. VI and show how these different approaches can lead to different answers about the same network. The correct approach depends on what kind of data analysis is being undertaken, which we discuss in our concluding Sec. VII.

II. GENERALIZED Z-MODULARITY

Modularity is usually defined as

$$Q_{\text{Newman}}(G,c)=\frac{1}{2E}\sum_{i}^{N}\sum_{j}^{N}\left(A_{ij}(G)-\frac{k_ik_j}{2E}\right)\delta(c(i),c(j)), \tag{1}$$

where N is the number of nodes in the network, $A_{ij}(G)$ is the adjacency matrix of the network G, $k_i=\sum_j A_{ij}$ is the degree of node i, $2E=\sum_i k_i$ is twice the number of edges, and $c(i)$ is the labeling function that returns the community of node i and defines a partition of the network.

The term $\frac{k_ik_j}{2E}$ is the approximate probability for two nodes to be connected in the configuration model of G, that is, if the degrees of the nodes are fixed and the connections randomized, the probability of an edge between i and j is $P_{ij}\simeq\frac{k_ik_j}{2E}$. This is valid only in the sparse limit, where no node has degree greater than $\sqrt{N}$ [23]. Note the correct expression, always valid, is well known [24] and should be used instead (see Sec. III). Other null models such as a random Erdős–Rényi graph where $P_{ij}=\frac{2E}{N(N-1)}$ or more complicated examples [25,26] could also be substituted. Allowing for an arbitrary null model gives

$$Q(G,P,c)=\frac{1}{2E}\sum_{ij}(A_{ij}-P_{ij})\delta(c(i),c(j)), \tag{2}$$

where P_{ij} is the probability of connecting i and j in the null.

Following [27], we can use a straightforward generalization of modularity to detect structures other than assortative communities. Defining the block variables

$$S_{ab}(G,c)=\sum_{i\in a,j\in b}A_{ij}(G), \tag{3}$$

$$T_{ab}(c)=\sum_{i\in a,j\in b}P_{ij}, \tag{4}$$

$$Q_{ab}=S_{ab}-T_{ab}, \tag{5}$$

where $\sum_{i\in a}\equiv\sum_{i|c(i)=a}$ is a convenient abuse of notation for summing over all nodes with label a. We have

$$Q(G,P,c)=\frac{1}{2E}\sum_{a}^{K}Q_{aa},$$

where K is the number of groups, expressing modularity as a sum over block variables. We introduce the $K\times K$ block matrix B_{ab}, which specifies the mesoscale structure we are searching for, and define

$$Q_{\text{block}}(G,P,B,c)=\frac{1}{2E}\sum_{a}^{K}\sum_{b}^{K}B_{ab}Q_{ab}. \tag{6}$$

For example, if B has ones on the diagonal and zeroes elsewhere, the above reduces to Q_{Newman}. As shown in [4,27,28], Q_{block} with different B matrices can be used to detect other mesoscale structures.

In hypothesis testing, the difference between observed and expected outcomes is usually measured relative to the variance of the null. The variance of the expected edge count between blocks a and b, assuming independent links, is

$$Var_{ab}(P_{ij})=\sum_{i\in a,j\in b}P_{ij}(1-P_{ij})=T_{ab}-U_{ab}, \tag{7}$$

with

$$U_{ab}=\sum_{i\in a,j\in b}P_{ij}^2. \tag{8}$$

We then define

$$Z_{ab}=\frac{Q_{ab}}{\sqrt{Var_{ab}}}=\frac{S_{ab}-T_{ab}}{\sqrt{T_{ab}-U_{ab}}}, \tag{9}$$

and finally

$$Z(G,P,B,c)=\frac{\sum_{ab}B_{ab}Z_{ab}}{\sqrt{\sum_{ab}B_{ab}^2}}=\frac{1}{K}\sum_{a}^{K}\sum_{b}^{K}B_{ab}Z_{ab}(G,P,c), \tag{10}$$

where the RHS uses the fact that generally we set $B_{ab}=\pm1$. Frequently, we do not need to know the actual value of Z, nor do we compare Z values for different Bs, so the overall normalization does not usually matter.

A similar idea was suggested before by [29], called Z-modularity. The difference between Z and [29]'s Z-modularity (apart from the possibility of nonassortative structures and different null models) is that we sum standardized residuals over each group, while [29] standardizes the total deviation Q.

Equation (10) is a definition, and we will use it to find partitions in Secs. V and VI showing its practical application. However, to motivate it theoretically and compare it to Q and [29]'s Z-modularity, consider a random partition of a network sampled from the null. In this case,

$$Q_{ab}=\sum_{i\in a,j\in b}(A_{ij}-P_{ij}),$$

is a sum of independent, binomial random variables with mean 0 and variance $P_{ij}(1-P_{ij})$. The central limit theorem says, under some mild conditions on P_{ij} (namely, that no single

term dominates the variance), the quantity $Z_{ab} = \frac{Q_{ab}}{\sqrt{Var_{ab}}}$ is distributed like a standard normal $\sim\mathcal{N}(0, 1)$ (for the distribution of Q over all partitions, see [28]). Equation (10) gives $Z \sim \mathcal{N}(0, 1)$, with this method of combining independent z-scores known as Stouffer's method in the literature on meta-analysis [30]. Contrast Z with Q or [29]'s Z-modularity, which both use $\sum_{ab} Q_{ab}$, so each term in this sum is $\sim\mathcal{N}(0, Var_{ab})$.

The value of Q_{Newman} evaluated on the optimal partition is sometimes used as a rule of thumb for measuring how modular a network is, though this is not a reliable way of assessing clustering quality [31]. Since we know $Z \sim \mathcal{N}(0, 1)$ on random partitions, a large value of Z on the optimal partition is a more reliable indicator of clustering quality; however, we advocate neither method and will describe a more robust approach in Sec. IV.

Finally, note that both [29]'s Z-modularity and Q_{Newman} sum over Q_{ab}. Q_{ab} measures how unexpected a partition is in absolute terms and $Q_{ab} \sim \mathcal{N}(0, Var_{ab})$. This means constructing groups with large variance and optimizing to find values in the right-hand tail leads to large values of Q_{ab}. Z_{ab}, on the other hand, measures how unexpected a partition is in relative terms, so each group contributes based on how (un)likely it is. Unlikeliness is what we are actually trying to maximize, not variance.

Unassigned nodes

A pattern to detect two communities would be

$$B = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix},$$

rewarding an excess of edges between nodes with the same label and a deficit of edges between nodes with different labels. If we add a row and columns of zeroes,

$$B = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

we now have three groups. The quality function Z with this block matrix rewards an excess of edges within and a deficit between the first two groups. Nodes in the last group do not contribute to the score. Therefore, nodes which can't be placed in either of the first two groups without decreasing the score can be placed in the third. We call this the unassigned group. As we will show later, see in particular Fig. 8, unassigned nodes are often peripheral ones, which have only a loose connection to the main bulk of labeled nodes.

We could use any values for B_{ab} to design a test statistic that searches for structures of interest. In this paper, for simplicity, we only use values $B_{ab} = \pm 1$ to specify structure and an extra row and column of zeroes when we want to allow unassigned nodes. See [4] for more discussion of possible patterns for the B matrix.

III. NULL MODELS

The test statistic Z depends on the null model P. Different choices affect what communities we can identify in a network, or indeed, if we can identify any at all. A null model assigns a probability $P(G)$ to a network G. This is often discussed in statistical mechanical language where we consider ensembles of networks [24,32]. If networks in the ensemble are subject to constraints, for example that they should have some fixed number of edges or a specified degree sequence, we determine $P(G)$ by maximizing the entropy subject to those constraints.

It is important to distinguish constraints that are satisfied exactly from constraints satisfied in an ensemble average, corresponding to the microcanonical and canonical ensembles, respectively. As argued by [23], we typically want to use the canonical ensemble, allowing for the fact that edges in our observed network may be imprecisely measured. When the number of constraints is extensive, which is the case in the configuration model and most of the models we consider, then the microcanonical and canonical ensembles are not equivalent [21].

We first write down the general recipe for computing the edge probability in the canonical ensemble and describe specializations for specific nulls below. The notation mostly follows [33], to which we also refer for more detailed derivations. Assuming independent Bernoulli edges with connection probability P_{ij}, the probability of generating a (binary, undirected, simple) network with adjacency matrix A is

$$P(A) = \prod_{i<j} P_{ij}^{A_{ij}} (1 - P_{ij})^{1-A_{ij}}. \tag{11}$$

The entropy is

$$S = -\sum_A P(A) \log P(A), \tag{12}$$

$$= -\sum_A P(A) \sum_{i<j} A_{ij} \log P_{ij} + (1 - A_{ij}) \log(1 - P_{ij}), \tag{13}$$

$$= -\sum_{i<j} P_{ij} \log P_{ij} + (1 - P_{ij}) \log(1 - P_{ij}), \tag{14}$$

where $\sum_A$ indicates a sum over the ensemble of all possible networks, and we used

$$P_{ij} = \sum_A P(A) A_{ij}. \tag{15}$$

The entropy can be maximized with constraints via the method of Lagrange multipliers,

$$\mathcal{S} = S + \sum_\mu \lambda_\mu (C_\mu(P) - \bar{C}_\mu), \tag{16}$$

where the constraints $C_\mu(P)$ are some functions that must have expected values equal to $\bar{C}_\mu$. Maximizing $\mathcal{S}$ gives

$$\log \frac{P_{ij}}{1 - P_{ij}} = \sum_\mu \lambda_\mu \frac{\partial C_\mu}{\partial P_{ij}}. \tag{17}$$

Using

$$\sigma(x) = \frac{1}{1 - e^{-x}}, \quad \sigma^{-1}(y) = \log \frac{y}{1 - y},$$

gives

$$P_{ij} = \sigma\left(\sum_\mu \lambda_\mu \frac{\partial C_\mu}{\partial P_{ij}} \right). \tag{18}$$

The parameters λ_μ are fixed by substituting P into the constraint equations and solving.

If the constraints are linear in P (as they will be in all of our examples), then

$$C_\mu = \sum_{ij} P_{ij} c_{\mu ij} \rightarrow \partial_{ij} C_\mu = c_{\mu ij} + c_{\mu ji}, \tag{19}$$

writing $\partial_{ij} C_\mu \equiv \frac{\partial C_\mu}{\partial P_{ij}}$ for brevity. It can be seen (by differentiating) that maximizing

$$\mathcal{L} = \sum_\mu \lambda_\mu \bar{C}_\mu - \sum_{i<j} \log\left(1 + \exp\left(\sum_\mu \lambda_\mu \partial_{ij} C_\mu\right)\right), \tag{20}$$

is equivalent to solving the constraint equations.

This form is almost equivalent to maximizing the likelihood of a Bernoulli random edge model. The log likelihood of such a model is

$$\log L(\lambda, \alpha | A) = \sum_{i<j} A_{ij} \log P_{ij} + (1 - A_{ij}) \log(1 - P_{ij}). \tag{21}$$

Substituting P_{ij} from Eq. (18) and using

$$\log \sigma(x) = x - \log(1 + e^x),$$
$$\log(1 - \sigma(x)) = -\log(1 + e^x),$$

gives

$$\log L(\lambda, \alpha | A) = \sum_{i<j} A_{ij} \sum_\mu \lambda_\mu \partial_{ij} C_\mu - \sum_{i<j} \log\left(1 + \exp\left(\sum_\mu \lambda_\mu \partial_{ij} C_\mu\right)\right), \tag{22}$$

which is equivalent to $\mathcal{L}$ when the constraints are in the form of Eq. (19) and

$$\bar{C}_\mu = \sum_{ij} A_{ij} c_{\mu ij}. \tag{23}$$

In this case, the constraint equations are

$$\sum_{ij} A_{ij} c_{\mu ij} = \sum_{ij} P_{ij} c_{\mu ij}. \tag{24}$$

A. Erdős–Rényi model

In this model, the number of edges is constrained to equal the number in the observed network

$$C(P) = \sum_{ij} P_{ij} = 2E. \tag{25}$$

Substituting into the above formulas, where in this case we can solve the constraint by hand, gives

$$P_{ij} = \sigma(\lambda) = p = \frac{2E}{N(N-1)}. \tag{26}$$

Testing against this null, we are asking the question: Is there community/mesoscale structure in the network beyond what would be expected in a random network with the same number of nodes and edges?

B. Configuration model

Here, the degrees are constrained to equal the observed degrees, on average. This gives N equations

$$\sum_j P_{ij} = \sum_j A_{ij}. \tag{27}$$

Substituting into the above formulas gives

$$P_{ij} = \sigma(\lambda_i + \lambda_j), \tag{28}$$

and the multipliers λ_i are found by maximizing

$$\mathcal{L} = \sum_i k_i \lambda_i - \sum_{i<j} \log(1 + e^{\lambda_i + \lambda_j}). \tag{29}$$

We will refer to this as the canonical configuration model. Testing against this null, we are asking: Is there community/mesoscale structure in the network beyond what would be expected from the node degrees?

C. Random dot product graph

The configuration model is a relatively weak null model. Many networks likely to be of interest in practice will have more structure than their degree distribution alone implies. A stricter null can be constructed using a random dot product graph (RDPG) [26]. We consider a low-rank approximation for the adjacency matrix

$$A \simeq V V^T, \tag{30}$$

where the N rows of V are d-dimensional vectors v_m. Standard results in linear algebra suggest that $v_m = \sqrt{\mu_m} e_m$, where e_m are the largest normalized eigenvectors of A with eigenvalues μ_m, will provide a good approximation.

Requiring that the null and the observed network have the same low-rank approximation gives Nd equations

$$\sum_j P_{ij} v_{jm} = \sum_j A_{ij} v_{jm}. \tag{31}$$

We also require that the edge count is the same,

$$\sum_{ij} P_{ij} = \sum_{ij} A_{ij}. \tag{32}$$

Substituting into the maximum entropy formulation, we get

$$P_{ij} = \sigma\left(\lambda + \sum_m \lambda_{mi} v_{jm} + \sum_m \lambda_{mj} v_{im}\right). \tag{33}$$

The multipliers can be found by maximizing

$$\mathcal{L} = \sum_{i<j} A_{ij}(\lambda + v_{jm}\lambda_{mi} + v_{im}\lambda_{mj}) - \sum_{i<j} \log(1 + e^{\lambda + \lambda_{mi} v_{jm} + \lambda_{mj} v_{im}}). \tag{34}$$

In this equation, the vectors v_{*m} are assumed to be known and fixed to equal the largest d eigenvectors of A. We maximize Eq. (34) to find the parameters λ_{mi}. This model is derived in an alternative way in [34]. Note that the canonical configuration model is a special case of the RDPG where we use a rank one approximation $v = \mathbf{1}$, a vector with all entries equal to 1.

Testing against this null, we are asking the question: Is there more community structure in the observed network than in any other network with the same low-rank approximation? It is important to note that failure to achieve significance with this null does not mean the absence of community structure; it means that whatever structure is there is already accounted for by the low-rank approximation. For example, the first two eigenvectors could indicate a strong community partition, with the higher modes providing no extra information. In this case, testing against the RDPG null with $d = 2$ would fail to achieve significance. This doesn't mean there are no communities in the network; it means two eigenvectors are enough to discover them. This model is for exploring the fine-structure of a network after accounting for the bulk communities, or determining when there is no fine-structure left.

D. Gravity model

Spatial networks often show many connections between spatially close nodes and few direct connections between distant nodes. One specific null for spatial data which encodes this tendency is the gravity model [25,35]. We start from the model of [36], which suggests, in the sparse limit,

$$P_{ij} = \delta w_i w_j f(d_{ij}), \tag{35}$$

where w_i is some known importance value associated with each node, e.g., population, $f(d)$ is the deterrence function which depends on the physical separation between nodes d_{ij}, and δ is a proportionality constant.

Requiring that the degree be proportional to the importance gives

$$\sum_j P_{ij} = \delta w_i. \tag{36}$$

Requiring that the number of edges be preserved gives

$$\sum_{ij} P_{ij} = \sum_{ij} A_{ij} = 2E, \tag{37}$$

which fixes δ since

$$\sum_{ij} P_{ij} = \delta \sum_i w_i = 2E, \tag{38}$$

so $\delta = 2E/W$ with $W = \sum_i w_i$.

If we ignore deterrence ($f(d_{ij}) = 1$), the degree constraint gives

$$P_{ij} = \sigma(\lambda_i + \lambda_j). \tag{39}$$

Often the identification $e^{\lambda_i} = \sqrt{\delta} w_i$ is made based on the sparse limit [25]. We could instead impose it as a constraint, by maximizing

$$\mathcal{L} = \sum_i \delta w_i \lambda_i - \sum_{ij} \log(1 + e^{\lambda_i + \lambda_j}), \tag{40}$$

as usual. This is not a maximum likelihood estimate, but it is a maximum entropy one. External constraints, i.e., not only involving network data A_{ij}, do not appear in the likelihood, Eq. (22), but they can fit in the maximum entropy model, Eq. (20). In practice, using this method, the values w_i cannot be too far from the degrees k_i; otherwise, it might be impossible to satisfy the constraints and find a maximum for Eq. (40), and we will ignore node importance values in the rest of this work, substituting $w_i = k_i$ and $\delta = 1$.

We introduce distance dependence using the constraint from [36] that the number of edges in each distance bin is fixed

$$\sum_{ij} P_{ij} I_b(d_{ij}) = \sum_{ij} A_{ij} I_b(d_{ij}), \tag{41}$$

where the bin indicator function for bin b

$$I_b(d_{ij}) = \begin{cases} 1 & d_{ij} \in b \\ 0 & \text{otherwise} \end{cases}, \tag{42}$$

and the bins are assumed to be given. The sum of edges is constrained automatically, since every edge is in some bin. These constraints together with the degree constraints give

$$P_{ij} = \sigma\left(\lambda_i + \lambda_j + \sum_b \lambda_b I_b(d_{ij})\right), \tag{43}$$

and the parameters are found by maximizing

$$\mathcal{L} = \sum_i \delta w_i \lambda_i + \sum_b \sum_{i<j} A_{ij} I_b(d_{ij}) \lambda_b - \sum_{i<j} \log\left(1 + \exp\left(\lambda_i + \lambda_j + \sum_b \lambda_b I_b(d_{ij})\right)\right). \tag{44}$$

Testing against this null, we are asking the question: Is there community structure in the network beyond what is implied by the node importances and the spatial distribution of edges?

IV. ALGORITHM

To perform community detection on some network G, we carry out the following procedure:

(1) Formulate a research hypothesis. Operationally, this means specifying the null model P and a target structure B, which also means choosing the number of groups K.

(2) Using Eq. (10), find

$$\hat{Z}(G, P, B) = \max_c Z(G, P, B, c), \tag{45}$$

by finding the partition of the observed network with the largest z-score.

(3) Generate N_P random networks G_p. In each network, connect each of the $N(N-1)/2$ unique pairs $i \neq j$ with probability P_{ij} [37].

(4) For all N_P generated networks, find

$$\hat{Z}_p(G_p, P, B) = \max_c Z(G_p, P, B, c). \tag{46}$$

(5) Estimate the right-tailed p-value by comparing $\hat{Z}$ to the $\hat{Z}_p$ values. If the p-value is less than some predefined threshold, say, $\alpha = 0.01$, reject the null and declare that the network has structure beyond the null. Otherwise, we can conclude that the observed network structure is found in many networks in the null ensemble. In this sense, we can say the structure is explained by the constraints.

The p-values computed here are properly pseudo p-values, since they are only estimates from a finite sample. One-sided pseudo p-values are computed by computing n_p, the number of $\hat{Z}_p$ values above $\hat{Z}$. The pseudo p-value is $\frac{n_p+1}{N_P+1}$. I will refer

to p-values for brevity, but they will always be computed by this method. I also note that it is the combination P and B which is rejected. One could fail to demonstrate significance for P_1 and B_1, but succeed for P_1 and B_2 or P_2 and B_1.

The overarching framework: generate samples from the null, compute a test statistic and compare against the observed data, is quite generic. One could, in principle, replace $\hat{Z}$ with any other function of the network. However, since $\hat{Z}$ incorporates the null, it is a particularly apt choice. A network which is unusual under P_1 need not be unusual under P_2. Likewise, different functions can be sensitive to different structures. For example, using $\hat{Z}$ with the ER null as the test statistic and sampling from the configuration null will not identify the same networks and communities as significant compared to using $\hat{Z}$ with the configuration null. The test statistic should be aligned to the null being tested, rather than one-size-fits-all.

We find the Lagrange multipliers by performing the maximisation using the LGBFS algorithm [38,39]. Each evaluation of the likelihood function and its derivatives is $O(N^2)$. Generating a network from the null is also $O(N^2)$. We find $\hat{Z} = \max_c Z(G, P, B, c)$ by label swapping with simulated annealing, as described in [27]. Each label swap proposal requires $O(N)$ operations, so each pass through the entire network is again $O(N^2)$.

The scaling is $O(N^2)$, rather than $O(E)$, because all of the P_{ij} values are potentially nonzero. In practice, many of them are very close to zero. Thus we could apply a threshold and sparsify the P matrix, giving us an operation count closer to $O(E)$. We did not do this here since it was not necessary for the small examples studied.

Without sparsification, $O(N^2)$ scaling means this method is not yet practical for very large networks. We leave speeding up the method for future work, but note that many real problems in network analysis are small to 'medium' data problems on, say, ∼100 to ∼1000 nodes. The examples below all run reasonably quickly on modest hardware [40], so for such problems, our approach is quite feasible.

V. ARTIFICIAL NETWORKS

A. Planted partition model

The planted partition model (PPM) [41] is a special case of the stochastic block model (SBM). Given a labeling function c on N nodes, the SBM generates a network with edge probability $P_{ij} = p_{c_i c_j}$, where the values p_{ab} control the probability of edges within or between groups. The PPM has all the diagonal probabilities equal to p_{in} and all the off diagonal probabilities equal to p_{out}, so $p_{ab} = p_{\text{out}} + \delta_{ab}(p_{\text{in}} - p_{\text{out}})$.

The degree-corrected stochastic block model (dc-SBM) [42] allows for more realistic degree distributions than the standard version. To generate networks from a degree-corrected planted partition model (dc-PPM), we sample N values t_i and block normalize them to get parameters θ_i. We will sample the t_i from a Pareto distribution with exponent γ. We then choose values of $\omega_{aa} = \omega_{\text{in}}$ and $\omega_{a\neq b} = \omega_{\text{out}}$ and use a Poisson model where the edge ij is added with probability $1 - \exp(-\theta_i\theta_j\omega_{g_i g_j})$. To be clear, we are only using the SBM to generate test networks; we are not trying to fit the parameters as is done in approaches to community detection based on the SBM [6]. We apply the steps of Sec. IV to study the detected community structure and compare it to the planted structure.

B. ER and canonical nulls

First, we study the standard PPM. When $p_{\text{in}} = p_{\text{out}}$, this is the ER model, so this makes an appropriate null. We fix $p_{\text{in}} = 0.8$ and, for various values $p_{\text{out}} \leqslant p_{\text{in}}$, construct $N_G = 100$ networks with three equally sized groups of 20 nodes using the PPM. On each of these networks, we determined pseudo p-values according to the procedure in Sec. IV, using the ER model as the null and performing $N_P = 100$ simulations per network. The data points in Fig. 1(a) show the results. When the value of p_{out} is low, community structure is strong. The observed network is not like an ER network with the same number of edges, and the p-value is low; in fact, we don't sample any networks with $\hat{Z}_p > \hat{Z}$, so $p = \frac{1}{N_P+1}$. When p_{out} is high, there is no community structure, the null is true, and the observed network is equivalent to a random one. In this case one would expect, and we do observe in practice, an approximately uniform distribution of p-values, giving an average p-value of 0.5 and an error bar $\sim\frac{1}{\sqrt{12}}\frac{1}{\sqrt{N_G}}$, as seen in Fig. 1(a). The distribution estimated from a finite sample of $N_G = 100$ is not exactly uniform because networks with strong community structure are extremely rare in the null; see [28] for more details on estimating distributions over null ensembles.

The value of p_{out} where the PPM transitions from having detectable to undetectable community structure is the Kesten-Stigum bound, which measures the signal-to-noise ratio [43,44],

$$(p_{\text{in}} - p_{\text{out}})^2 = \frac{K p_{\text{in}} + K(K-1)p_{\text{out}}}{N}. \tag{47}$$

Solving gives the value of p_{out} where we observe the p-value increase sharply, indicating the planted structure is not statistically significant.

Figure 1(b) shows a problem case identified by [22] where traditional modularity maximization overfits. The network is an ER random graph with $N = 30$ nodes and connection probability $p = 0.1$ where we have added extra edges to the first N_{clique} nodes to make a clique. Call this the planted clique model. We again use the ER null $N_G = 100$, $N_P = 100$, and this time,

$$B = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix},$$

to detect one community among unassigned nodes. Before adding the clique, the mean degree is $\bar{k} \sim 3$, and the standard deviation is $\sigma \sim 0.4$. Figure 1(b) shows that when the degree of the nodes in the clique $N_{\text{clique}} - 1$ is two sigma or more greater than $\bar{k}$, we can identify significant community structure. The inset shows that the clique nodes are the ones grouped in a community by maximizing Z. Figure 2 shows the same experiment using the dc-PPM with $\omega_{\text{in}} = 150$ and $\gamma = 3$. We use the canonical configuration model as the null and $N_G = 100$, $N_P = 100$ as before. This means we are looking for clustering that is not explained by the degree sequence and find that significant community structure cannot

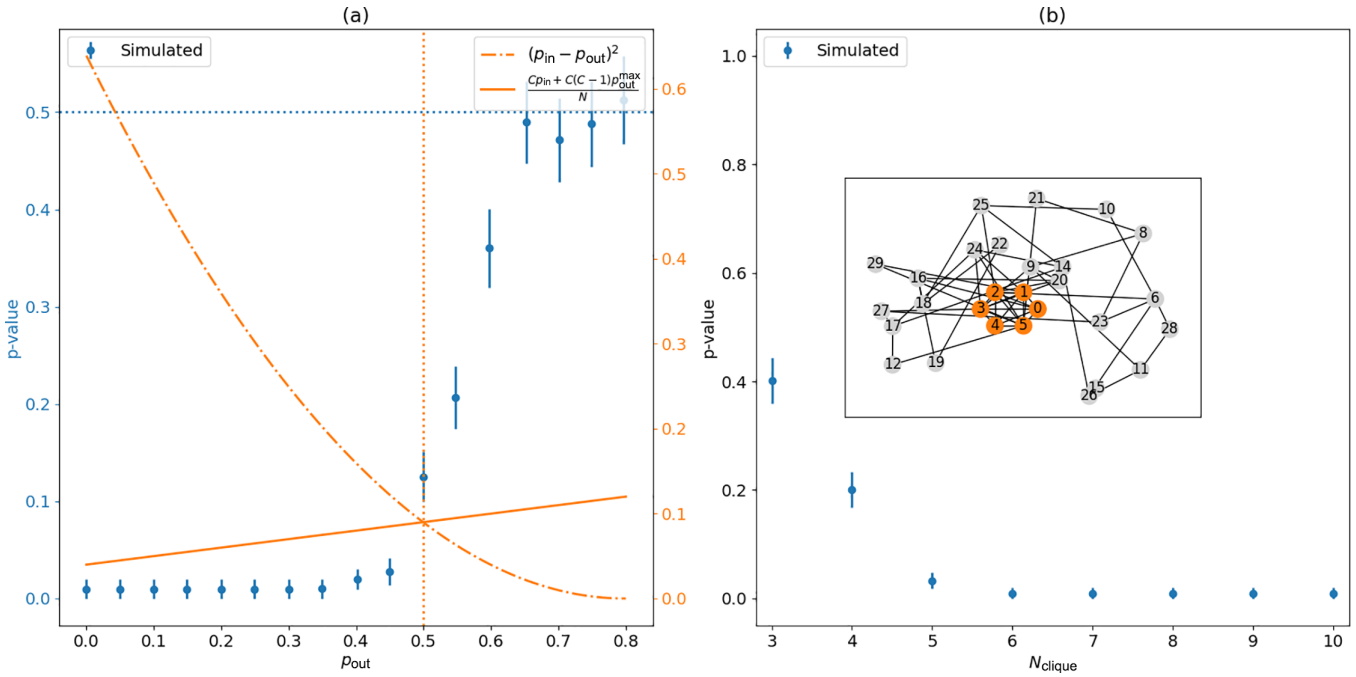


FIG. 1. (a) p-values for detection of a three-community PPM under the ER null model with $p_{in} = 0.8$. The vertical dashed line shows where the Kesten-Stigum bound is broken. The horizontal dashed line shows a p-value of 0.5. (b) p-values for ER model+clique; inset figure shows the detected community for an example network with $N_{clique} = 6$. Gray nodes are unassigned. $N_G = 100$, $N_P = 100$ used for (a) and (b).

be detected above the Kesten-Stigum bound. This does not mean the planted partition could not be recovered by some method. Although the transitions are sharp, there is still a weight of evidence for structure in the observed graph, with a p-value <0.5, so a partition at least correlated with the planted one could be recovered [43,45]. A nonsignificant result (say, $p > 0.01$) means that there is nothing unusual about this graph in the canonical configuration model. Many networks with the same degree sequence will have a similar community structure. See [4] for other examples where hypothesis testing and an information-theoretic approach give different views on the same network.

C. Random dot product graph null

To test the RDPG null, we generate a network with 100 nodes and five equally sized communities using the dc-PPM with $\omega_{in} = 500$, $\omega_{out} = 5$, and θ chosen by sampling from a Pareto distribution with $\gamma = 3$. We use the RDPG null, Eq. (33), with v_m equal to the largest d eigenvectors to detect communities. The p-values labeling the top row of Fig. 3 show that for $d = 1, 2, 3$, there is still evidence for unusual community structure in the observed network compared to the null. For $d = 4, 5$, there is no strong evidence for more community structure in the observed network than in a random sample from the null. This implies that four eigenvectors are enough to resolve the clusters in this network.

The bottom row of Fig. 3 shows an example network generated from each null. As the force-directed layout and the colors imply, there is weak evidence for five clusters for the $d = 1, 2, 3$ networks while the community structure for $d = 4, 5$ is obvious. To be clear, the high p-values for $d \geqslant 4$, which are averages over $N_G = 100$ different networks, do not mean that the observed network doesn't have community structure. This means that any network with the same rank-4 approximation, like the ones in the bottom row, has just as much structure as the observed network.

D. Gravity null

We use the PPM with $p_{in} = 0.5$ and $p_{out} = 0.025$ to generate $N_G = 100$ networks with two blocks and 30 nodes per block. We then locate each of the blocks by assigning a random Gaussian position, $(x, y) \sim (\mathcal{N}(\mu_x, \sigma), \mathcal{N}(0, \sigma))$, where $\mu_x = 0$ for the first block and $\mu_x = 1$ for the second. As shown in Fig. 4, when σ is small, the blocks are spatially well separated, and the null predicts many edges between nearby nodes and few between distant nodes. Community detection with the gravity null gives a high p-value, so we fail to find significant clustering beyond what is expected given the distance between nodes. On the other hand, when σ is large, the two planted communities overlap spatially, and the spatial position alone does not imply strong clustering into two groups. Thus the excess of connections between the groups is more than the gravity model would suggest, and the planted communities are significant.

VI. COMPARISON WITH SBM INFERENCE

An increasingly popular approach to network community detection is Bayesian inference [6,46,47]. The basic idea is to maximize the posterior $P(c|A)$, which expresses the probability that the partition c generated the data A according to an underlying generative model, usually some variant of the SBM. Finding the partition which maximizes the posterior probability of the data is equivalent to minimizing the

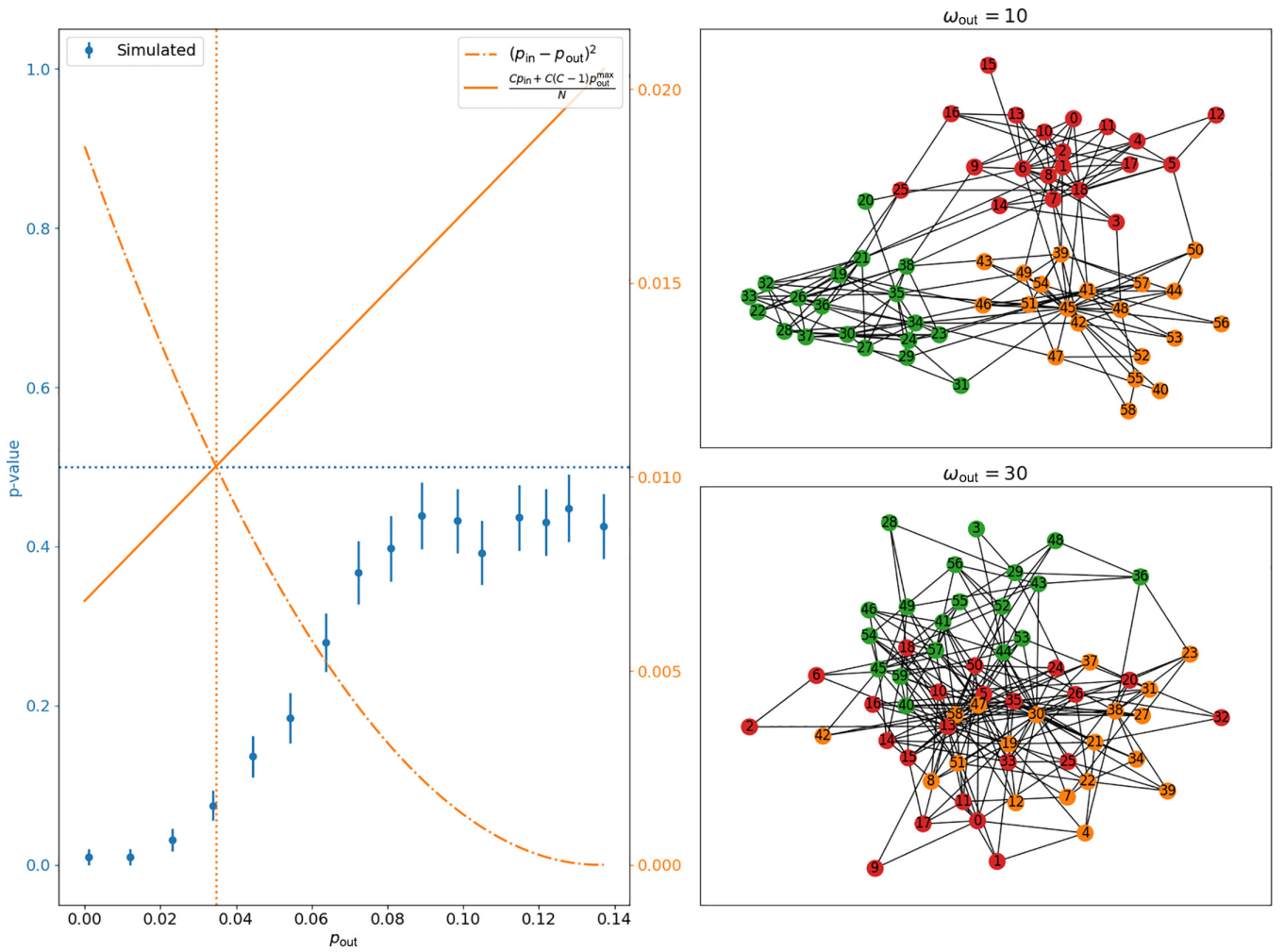


FIG. 2. Left shows p-values for detection of a three-community dc-PPM against the canonical configuration model. $N_G = 100$, $N_P = 100$, model parameters $\gamma = 3$, $\omega_{in} = 150$. The vertical dashed line shows where the Kesten-Stigum bound is broken. The horizontal dashed line shows a p-value of 0.5. On the right are community partitions found by maximizing Z, when the intercommunity edges are sparse $\omega_{out} = 10$, and when they are denser $\omega_{out} = 30$. Communities are planted so that indices 0 to 19 are in community 1, 20 to 39 in 2, and 40 to 59 in 3.

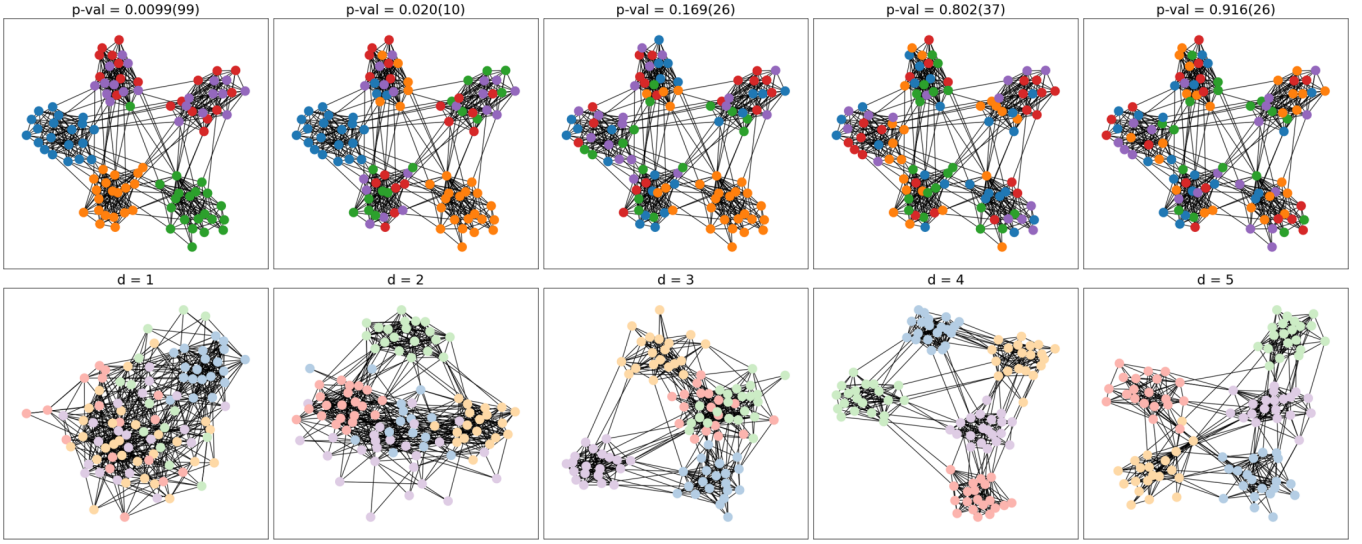


FIG. 3. Top row shows a network generated with the dc-PPM, with communities detected using the RDPG null using the top $d = 1, 2, \ldots 5$ eigenvectors as constraints. Bottom row shows an example network generated by sampling edges from the corresponding null and coloring based on communities detected with the canonical configuration model. $N_G = 100$, $N_P = 100$ model parameters $\omega_{in} = 500$, $\omega_{out} = 5$, $\gamma = 3$.

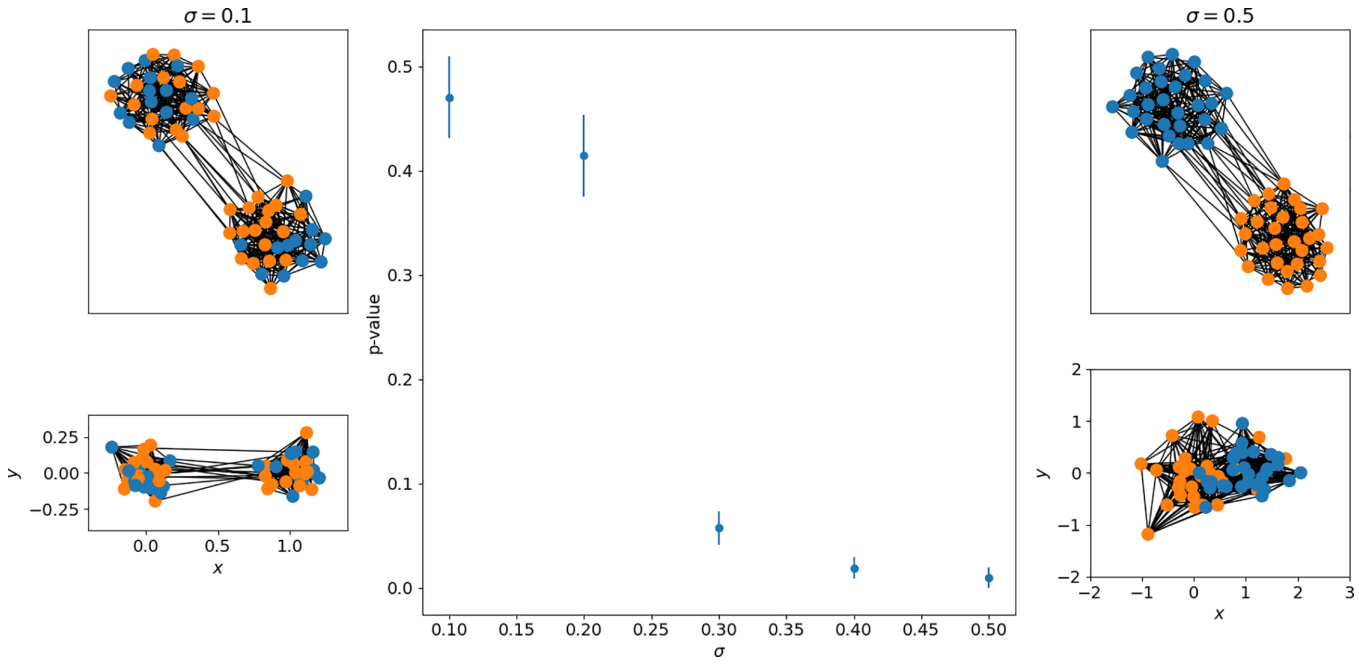


FIG. 4. In the left and right panels, the top shows the network's topological layout and the bottom shows the spatial layout. The central scatter plot shows p-values versus σ. When σ is small, the spatial positions imply clustering will happen in the gravity model. Thus there is nothing unusual about the planted communities; clustering is expected when nodes are close. When σ is large, the gravity model implies less clustering within the groups than we have planted, so significant community structure can be detected. $N_G = 100$, $N_P = 100$ model parameters $p_{\text{in}} = 0.5$, $p_{\text{out}} = 0.025$, spatial positions sampled from Gaussian distributions centered at (0,0) and (1,0).

description length, the number of bits required to specify the data given the model [48]. The best partition according to this approach is the maximum *a posteriori* (MAP) estimator $\hat{c} = \arg\max_c P(c|A)$. This partition also minimizes the description length; hence, this approach is often referred to as the minimum description length (MDL) principle. For more details, see the cited references.

This approach, broadly speaking, answers the question: If my data were generated by an SBM, what were its parameters? The parameters usually of most interest are the block membership labels, which partition the nodes. The SBM and its variants are quite flexible, and many networks that interest network scientists could be plausibly generated by one. The MDL principle therefore represents a reasonable approach to unsupervised clustering on networks.

The approach described in Sec. IV is frequentist rather than Bayesian inference. Instead of estimating distributions of model parameters, as with Bayesian methods, we consider our observed network as one possibility chosen from a large set of potential networks generated by the null. If our null model has a very low probability of generating the observed network, we can rule out the possibility that the observed network came from the null. Compared to a completely unsupervised learning task or model fitting, this approach is most suitable for answering specific questions about the data, with assumptions made explicit through the choice of null.

Another contrast between our procedure and the Bayesian approach is that the Bayesian method uses hard constraints [49]. This means when computing $P(c|A)$ by

$$P(c|A) = \frac{P(c,A)}{P(A)}. \tag{48}$$

$P(c,A)$ is marginalized using

$$P(c,A) = \sum_{\pi} P(c,A,\pi) = P(c,A,\hat{\pi}), \tag{49}$$

where π represent model parameters. The formula above means that instead of summing over all possible model parameters weighted by probability, only one term is used to evaluate the sum, the term using the parameters $\hat{\pi}$ that correspond to the values computed on the observed network, i.e., sums are over the microcanonical ensemble.

Ensemble equivalence means, in the limit of large N, averages computed on the microcanonical and canonical ensembles agree. This does not hold in general. In the constraint-based, entropy maximization formulation, canonical and microcanonical ensembles are not equivalent when the number of constraints is extensive (proportional to N) [21]. [23] shows that for the standard SBM we do have ensemble equivalence, but not for the dc-SBM, which is typically used in Bayesian community detection. The canonical approach is suitable when there is the possibility of noise in the data, e.g., missing links. Since this is the case for most network data of interest, the canonical ensemble is the appropriate one. We also note recent work [50] showing that when an extensive number of constraints are enforced, different choices of priors can lead to large differences in description length, of order $\Theta(N)$.

The remainder of this section will compare the Bayesian and frequentist approaches on some well-known networks.

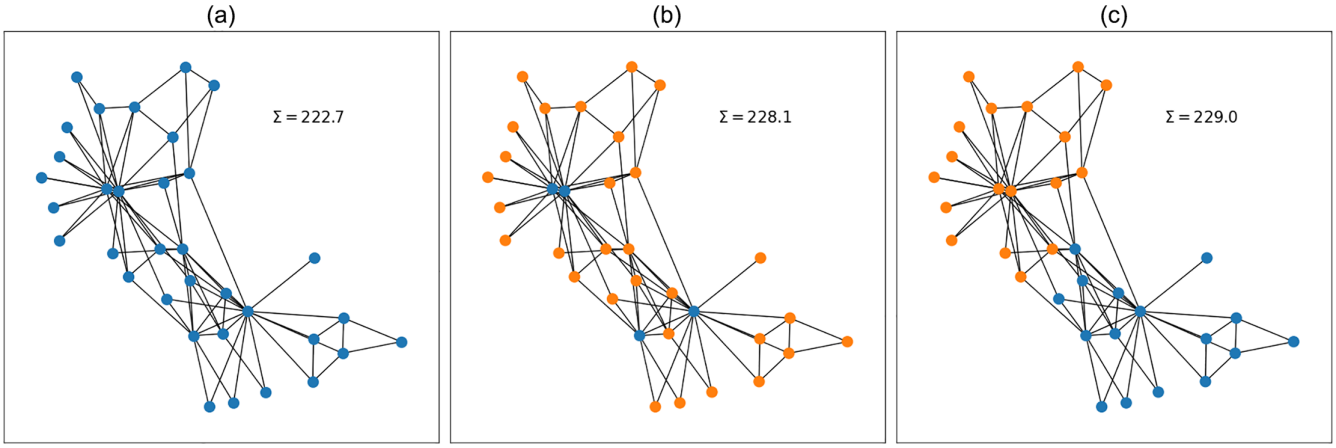


FIG. 5. Partitions of the karate club network found with dc-SBM inference. Left is the MAP estimator, indicating no communities. Two other high-probability partitions are shown with the description length Σ indicated (measured in nats, $\log_e$).

A. Karate club

Figure 5 shows the well-known karate club graph [51], a network of social interactions between members of a karate club just prior to a split caused by a conflict between the club's instructor and president. The analysis of this network from [47] is repeated using the graph-tool library [6]. The MAP estimate, using the dc-SBM as the underlying model, is that no community structure is supported; the karate club is consistent with a random graph with the specified degree sequence. If we soften the MDL principle (the small description length principle perhaps) and force $K = 2$ communities, we find that a leader-follower partition and a two-community split are high-probability partitions. Given these three rather different possibilities, it is not clear what one should conclude about the community structure of the karate club. Following the MDL principle strictly, there is no community structure, contradicting what is known about the social interactions encoded by the network. Being less strict and looking at other high-probability partitions, we have two completely different labelings of the network with similar description lengths.

Our analysis follows the procedure in Sec. IV. We want to know if the social interactions encoded by the network reflect the conflict between the instructor and president, or if they are simply because these individuals have a lot of friends. We translate this to the question: Does the observed network have a stronger community structure than any other network with the same degree sequence? Here, strength is measured by Z and the null ensemble enforces the degree constraint on average.

Doing the optimization and simulation, the answer is yes. Figure 6(a) shows that in $N_P = 1000$ samples from the null, we never observe a network with such strong community structure. This holds even if we allow an unaligned partition by appending a row and column of zeroes to B as in Sec. II A, the optimal partition does not use the unaligned label; it is better, in terms of optimal Z, to put all nodes in one or the other community.

We can experiment with other structures. A bipartite structure

$$B = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix},$$

is not significant, nor is a repulsive partition

$$B = \begin{pmatrix} -1 & 0 \\ 0 & 0 \end{pmatrix},$$

shown in Fig. 6(b). Interestingly, in this case the observed data has significantly smaller $\hat{Z}$ than a random network with this degree sequence. So this network is unusually difficult to split into a group which does not interact and a group which interacts indiscriminately. This strongly implies the feud is not restricted to just a few key individuals.

We can still refine this picture. Using a double core-periphery (CP) [2] block pattern,

$$B = \begin{pmatrix} 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & -1 \end{pmatrix}.$$

In 1000 samples from the null, we find no networks with equal or stronger double CP structure. The partition is shown in Fig. 6(c) and suggests individuals who may only be peripherally involved in the conflict.

B. Political blogs

Figure 7(a) shows a network of links between political blogs (treated as undirected) [52]. The force-directed layout agrees with intuition that there should be two main clusters (left- and right-wing blogs), although they appear to be strongly overlapping. The MAP estimate suggests 19 communities, which are hard to interpret without any additional metadata.

Figure 8(a) shows a vector representation of each node using the first two eigenvectors of the adjacency matrix. We can cluster this vector data using a standard algorithm [7] and use the labels to color the nodes of the network, as in Fig. 8(b), suggesting the left/right split and a large periphery. We can reproduce a similar labeling by maximizing Z under the ER null, with the structure matrix

$$B = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

shown in Fig. 8(c).

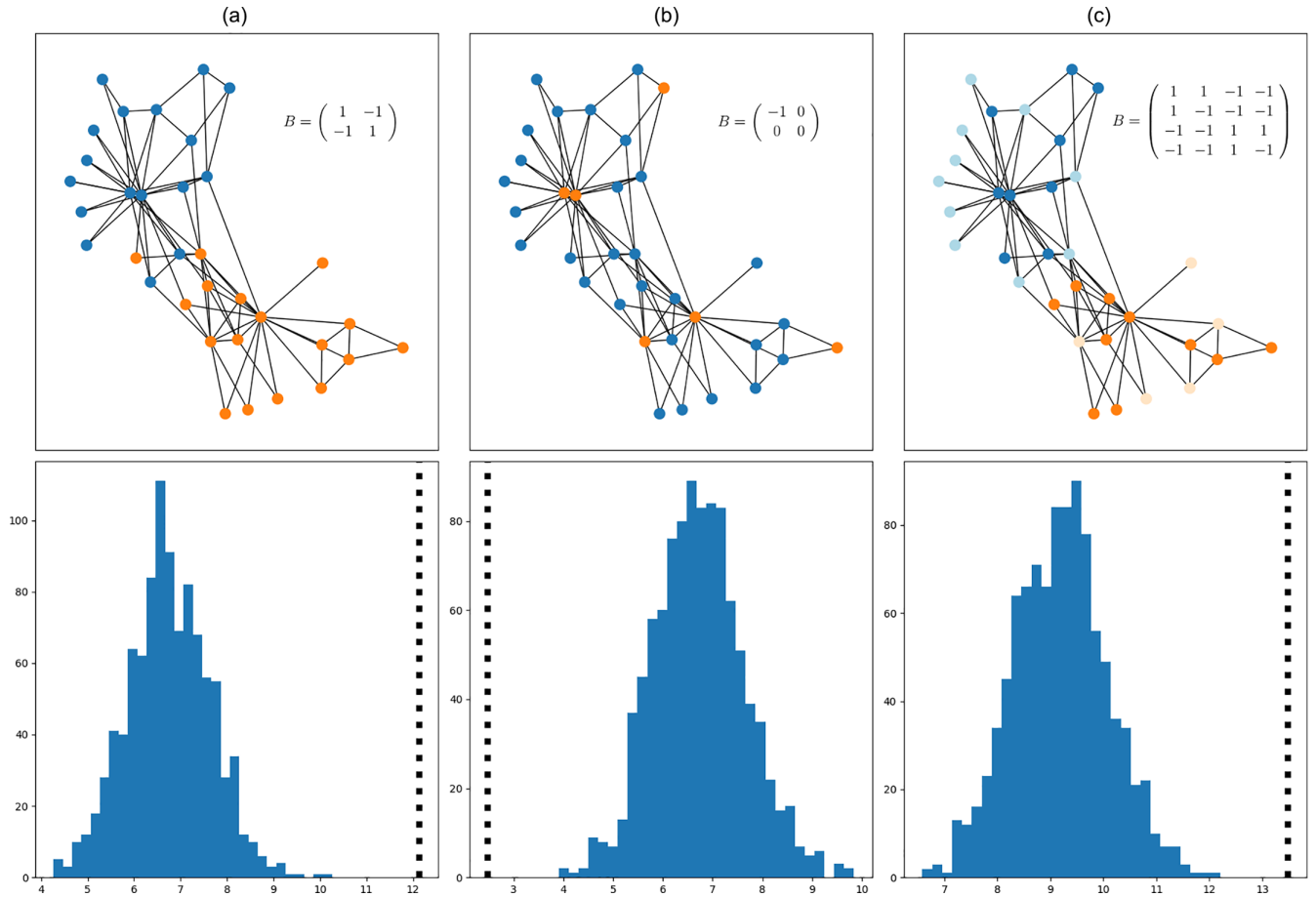


FIG. 6. (a) Partitions of the karate club network found from optimizing Z with the canonical configuration model and different B. Left shows a two-community partition (b), a repulsive partition (c), and a double core-periphery. Bottom shows the distribution of Z values from the null, with the dashed line corresponding to the value $\hat{Z}$ computed on the observed network $N_P = 1000$.

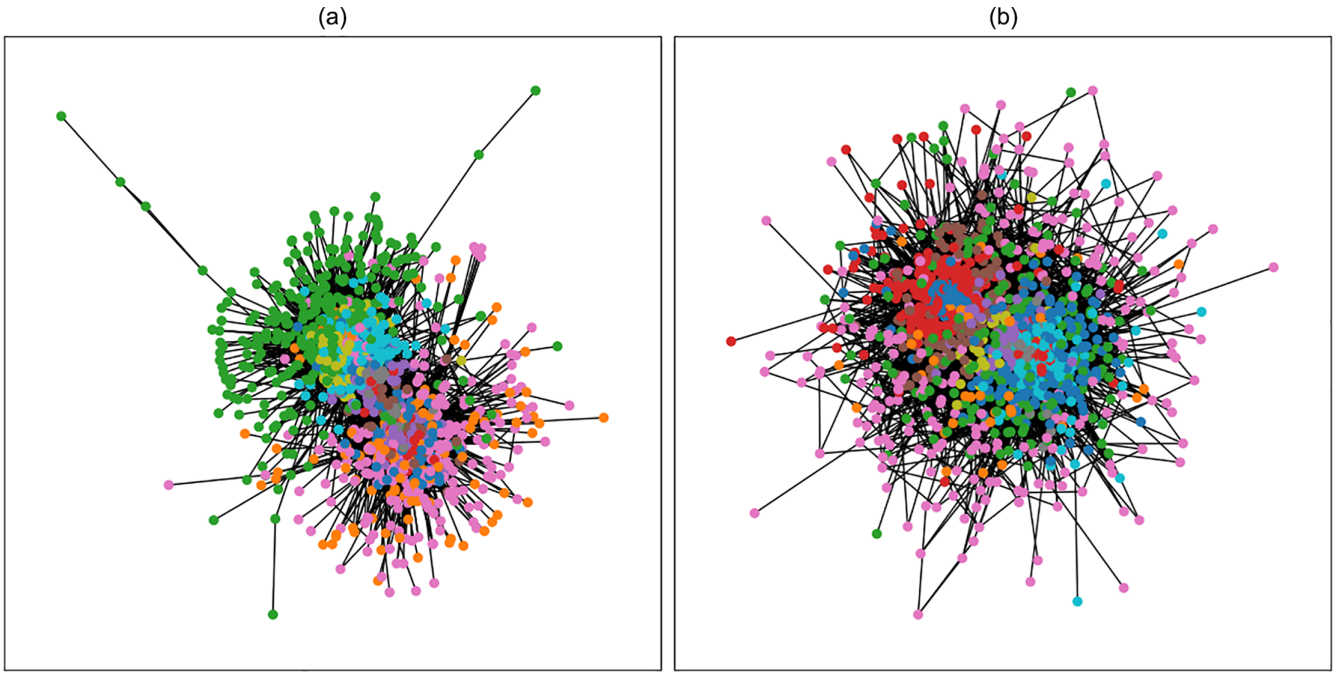


FIG. 7. (a) SBM-based minimum description length partition of the political blogs network. There are 19 distinct groups. (b) MDL partition of a random network from the two-eigenvector RDPG null model. This has 15 groups.

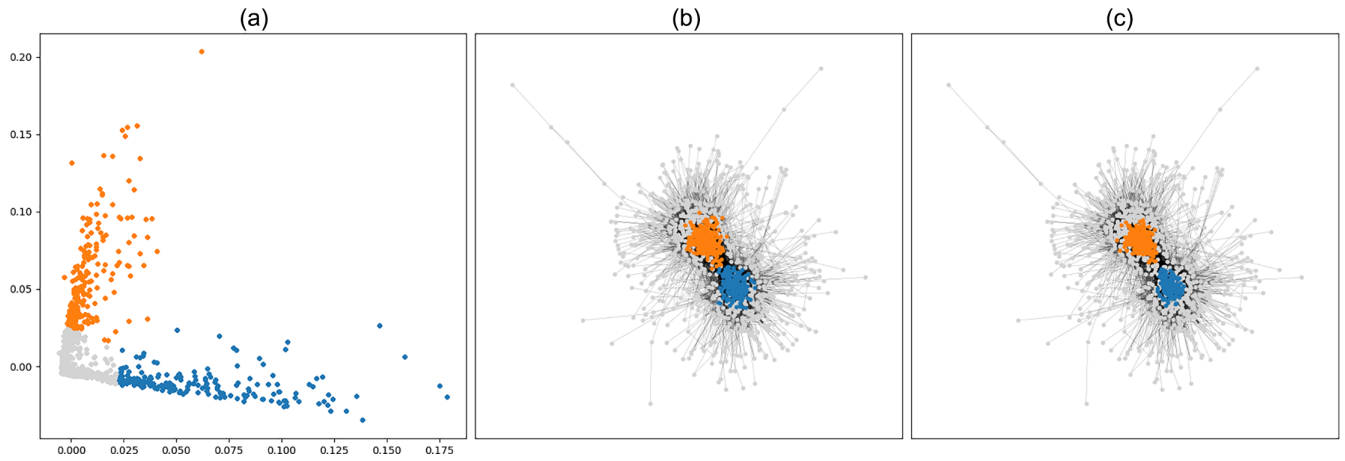


FIG. 8. (a) Vector representation of the political blogs data using the two largest eigenvectors. Colors derived from a vector-based clustering algorithm. (b) Political blogs network, colored based on eigenvector clustering. (c) Political blogs network, colored based on labels maximizing Z, using the ER null and B equal to a two-community plus unaligned (gray).

We will now construct an RDPG model which fixes the first two eigenvectors, i.e., A and P have the same rank-2 approximation. We can use this to generate networks which have the same rank-2 approximation as the political blogs data. Figure 7(b) shows such a network along with its MAP partition, obtained using graph-tool. This suggests the generated network has a partition of 15 groups. Given the network is randomly generated and contains no more information than Fig. 8(a), the MAP partition is most likely spurious, or at least very difficult to interpret. As in Sec. V C, we can explore how many vectors are required to explain all of the community structure in the political blog network. We construct P_{ij} as in Sec. V C using $d = 2 \ldots 20$ eigenvectors and measure the significance of partitions found using the two-community plus unaligned structure matrix B. Figure 9(a) shows that $\sim$16 eigenvectors are enough to explain all the structure, i.e., this is the point at which networks from the null become statistically indistinguishable, as measured by $\hat{Z}$, from the observed network. Figure 9(b) uses the top 20 eigenvectors to cluster the data, and Fig. 9(c) shows these cluster labels applied to the network. This gives two large communities and a few unaligned nodes. This is much more in line with intuition, and with the metadata on affiliation presented in [52], than the MDL partition into 19 groups. It can also answer the most likely questions we have about the data, i.e., is this blog left-wing, right-wing, or neither?

VII. DISCUSSION

Most discussions of community detection frame it as exploratory data analysis. We have a network and are completely in the dark as to what kinds of groups or structures it might contain. We contend that this is not the only, or even the most common, use case for community detection. We often have metadata about the nodes, e.g., a social network of students where we know the courses the students are taking. We also often have specific questions about the network, e.g., do friendship groups reflect political party affiliation? One

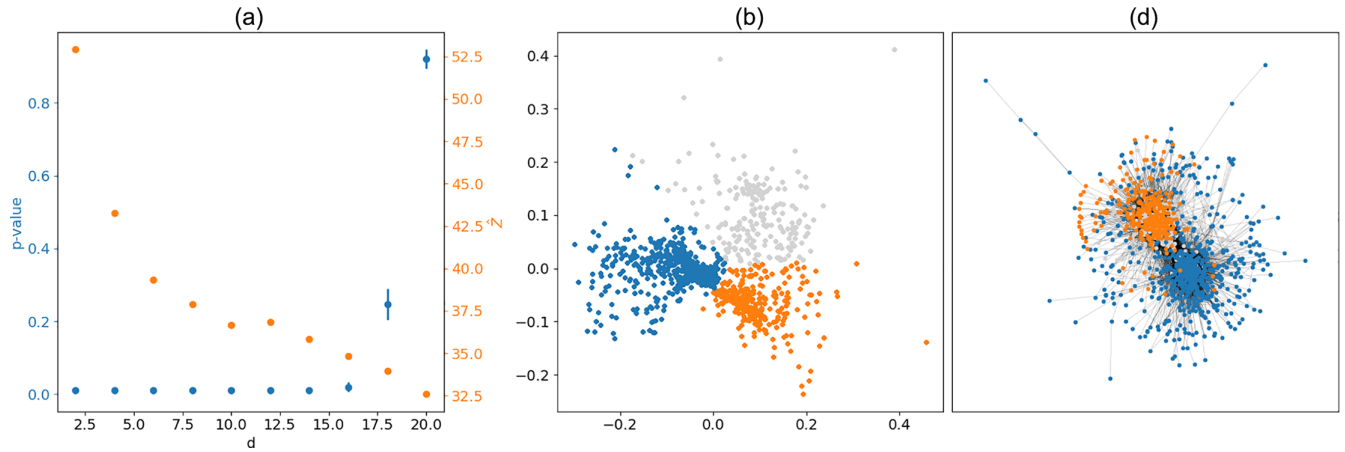


FIG. 9. (a) p-value of the two-community plus unaligned partition using an RDPG null with d eigenvectors. Orange shows the value of $\hat{Z}$, blue shows the p-values. $N_P = 100$. (b) Clustering nodes based on the highest 20 eigenvectors. Data is projected to 2D using the Isomap algorithm [53] for the purposes of visualization, but is clustered in 20 dimensions. (c) Political blogs network, colored based on labels determined in (b).

approach is to incorporate this metadata into the community detection algorithm [54,55]. In this work, rather than guiding the exploration with metadata, we suggest a move from exploratory to confirmatory data analysis and formal hypothesis testing.

For exploratory analysis, one should define what is meant by a community, which is what different algorithms do, explicitly or implicitly, e.g., communities are connected by bridges [56], communities are places where random walkers get stuck [57], communities allow us to effectively compress the network [47], etc. In this paper, a community is defined by maximizing Z, so for us a community partition maximizes the z-score, the difference between observed and expected edge counts relative to the variance. Inequivalent notions of community might not necessarily reproduce the same partitions. This does not mean one algorithm is always better or results should be averaged to form a consensus. It means different labeling algorithms are looking for different things and the analyst must be precise about what they mean by a community. On the other hand, this does not mean that all algorithms are equally valid, and some approaches, in particular standard modularity maximization [22], have serious problems and should be avoided when good alternatives are available.

However, finding labels using some algorithm is not the final step. In this paper, we have discussed performing confirmatory data analysis by comparing the observed network to other networks generated by a null model. We particularly emphasize using null models which are mathematically sound, i.e., constructed from correctly normalized probabilities that sample the canonical ensemble. Nulls should also be sufficiently powerful. In medical testing, it is easier to demonstrate a significant effect by comparing a new medicine to a placebo than by comparing it to the current best treatment. Similarly for community detection, some structure might be significant under, say, an ER null, but not under a configuration null. Or it might be significant under both but not under an RDPG null fixing the first few eigenvectors.

The examples shown in Secs. V and VI demonstrate this and counter the contention of [22] that hypothesis testing approaches to community detection always demonstrate significance and offer little insight. As also shown in Secs. V and VI, the partition found depends on the null. Another excellent example (albeit using modularity) is given by [36], where using the configuration model communities correspond to geographic regions, and using a spatial null, communities correspond to linguistic groups. The correct partition depends on the question you are asking. Such spatial null modeling fits well in our framework [35] and will be explored more in future work.

Of course, hypothesis testing is not a panacea to be used uncritically. Doing so causes many well-known issues, e.g., [58]. Most relevant here are so-called researcher degrees of freedom [59], where someone attempting to detect community structure could try many different P and B matrices until they achieve significance. One could correct this, lowering the significance threshold using standard methods like Bonferroni correction [60] or false discovery rate [61]. In cases where community structure and membership *per se* are the primary research findings, researchers should use pre-specified hypotheses or correct for multiple testing. However, it is often the case that one wants to compare discovered labels with metadata, or use community structure as part of a wider analysis of some network. In such cases, I advocate eschewing formal significance levels and, following more recent practice [62–64], using p-values as indicative of an interesting result, which should be considered alongside metadata and other statistics.

This perspective fits with the recognition that exploratory and confirmatory data analysis are complementary. Our approach requires formulating precise research questions. Mathematically, this means picking P, B and K. Specific research hypotheses, metadata, or exploratory data analysis, perhaps involving unsupervised community detection, are all reasonable ways to suggest choices of P, B, and K. Exploratory data analysis is important and can give useful insight into network data. But, as in most fields of science, strong conclusions require carefully formulated hypotheses and quantified uncertainty. This applies just as much to network community detection as it does to the measurement of any other quantity. Rather than relying on the outputs of some black-box algorithm, or worse, averaging together the outputs of many black boxes, practitioners should be encouraged to make testable and concrete hypotheses about the structure of their networks. Methods like those presented here can then test such hypotheses and generally move us towards more rigorous community detection.

ACKNOWLEDGMENT

I would like to thank Federico Botta and two anonymous reviewers for their comments on this manuscript.

DATA AVAILABILITY

There are no publicly available research data or software supporting this manuscript. Requests for further information or data should be sent to the authors.

[1] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, Defining and identifying communities in networks, Proc. Natl. Acad. Sci. USA **101**, 2658 (2004).

[2] S. Kojaku and N. Masuda, Finding multiple core-periphery pairs in networks, Phys. Rev. E **96**, 052313 (2017).

[3] L. Chen, Q. Yu, and B. Chen, Anti-modularity and anti-community detecting in complex networks, Inf. Sci. **275**, 293 (2014).

[4] R. Arthur, Detectability constraints on meso-scale structure in complex networks, PLoS ONE **20**, e0317670 (2025).

[5] C. X. Liu, T. J. Alexander, and E. G. Altmann, Nonassortative relationships between groups of nodes are typical in complex networks, PNAS nexus **2**, pgad364 (2023).

[6] T. P. Peixoto, Hierarchical block structures and high-resolution model selection in large networks, Phys. Rev. X **4**, 011047 (2014).

[7] U. von Luxburg, A tutorial on spectral clustering, Stat. Comput. **17**, 395 (2007).

[8] S. White and P. Smyth, A spectral clustering approach to finding communities in graphs, in *Proceedings of the 2005 SIAM International Conference on Data Mining (SDM)* (SIAM, Philadelphia, PA, 2005), pp. 274–285.

[9] J. Kleinberg, An impossibility theorem for clustering, in *Proceedings of the 16th International Conference on Neural Information Processing Systems*, NIPS'02 (MIT Press, Cambridge, MA, 2002), pp. 463–470.

[10] U. Von Luxburg, R. C. Williamson, and I. Guyon, Clustering: Science or art? JMLR: Workshop Conf. Proc. **27**, 65 (2012).

[11] S. Kojaku and N. Masuda, A generalised significance test for individual communities in networks, Sci. Rep. **8**, 7351 (2018).

[12] A. Lancichinetti, F. Radicchi, and J. J. Ramasco, Statistical significance of communities in networks, Phys. Rev. E **81**, 046110 (2010).

[13] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato, Finding statistically significant communities in networks, PLoS ONE **6**, e18961 (2011).

[14] M. E. J. Newman, Modularity and community structure in networks, Proc. Natl. Acad. Sci. USA **103**, 8577 (2006).

[15] C. P. Massen and J. P. K. Doye, Thermodynamics of community structure, arXiv:cond-mat/0610077.

[16] E. Yanchenko and S. Sengupta, A generalized hypothesis test for community structure in networks, Net. Sci. **12**, 122 (2024).

[17] J. Zhang and Y. Chen, A hypothesis testing framework for modularity based network community detection, Stat. Sin. **27**, 437 (2017).

[18] M. Ankerst, M. M. Breunig, H. P. Kriegel, and J. Sander, OPTICS: Ordering points to identify the clustering structure, ACM Sigmod Record **28**, 49 (1999).

[19] R. J. G. B. Campello, D. Moulavi, and J. Sander, Density-based clustering based on hierarchical density estimates, in *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (Springer, Berlin, 2013), pp. 160–172.

[20] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)* (AAAI Press, Reston, VA, 1996), pp. 226–231.

[21] T. Squartini, J. De Mol, F. Den Hollander, and D. Garlaschelli, Breaking of ensemble equivalence in networks, Phys. Rev. Lett. **115**, 268701 (2015).

[22] T. P. Peixoto, *Descriptive vs. Inferential Community Detection in Networks: Pitfalls, Myths and Half-Truths* (Cambridge University Press, Cambridge, 2023).

[23] D. Garlaschelli, F. Den Hollander, and A. Roccaverde, Ensemble nonequivalence in random graphs with modular structure, J. Phys. A: Math. Theor. **50**, 015001 (2017).

[24] J. Park and M. E. J. Newman, Statistical mechanics of networks, Phys. Rev. E **70**, 066117 (2004).

[25] D. Garlaschelli and M. I. Loffredo, Fitness-dependent topological properties of the world trade web, Phys. Rev. Lett. **93**, 188701 (2004).

[26] S. J. Young and E. R. Scheinerman, Random dot product graph models for social networks, in *International Workshop on Algorithms and Models for the Web-Graph* (Springer, Berlin, 2007), pp. 138–149.

[27] R. Arthur, Discovering block structure in networks, Physica A **613**, 128527 (2023).

[28] R. Arthur, Exploring network structure with the density of states, Physica A **674**, 130742 (2025).

[29] A. Miyauchi and Y. Kawase, *Z*-score-based modularity for community detection in networks, PLoS ONE **11**, e0147805 (2016).

[30] A. B. Owen, Karl Pearson's meta-analysis revisited, Ann. Stat. **37**, 3867 (2009).

[31] J. Reichardt and S. Bornholdt, When are networks truly modular? Physica D **224**, 20 (2006).

[32] G. Bianconi, Entropy of network ensembles, Phys. Rev. E **79**, 036114 (2009).

[33] G. Bianconi, Maximum entropy models for complex networks, lecture slides, *London Taught Course Centre* (Queen Mary University of London, 2022).

[34] L. J. O'Connor, M. Medard, and S. Feizi, Maximum likelihood embedding of logistic random dot product graphs, in *Proceedings of the AAAI Conference on Artificial Intelligence* (AAAI Press, Reston, VA, 2020), pp. 5289–5297.

[35] M. Di Vece, D. Garlaschelli, and T. Squartini, Gravity models of networks: Integrating maximum-entropy and econometric approaches, Phys. Rev. Res. **4**, 033105 (2022).

[36] P. Expert, T. S. Evans, V. D. Blondel, and R. Lambiotte, Uncovering space-independent communities in spatial networks, Proc. Natl. Acad. Sci. USA **108**, 7663 (2011).

[37] T. Squartini, R. Mastrandrea, and D. Garlaschelli, Unbiased sampling of network ensembles, New J. Phys. **17**, 023052 (2015).

[38] D. C. Liu and J. Nocedal, On the limited memory BFGS method for large scale optimization, Math. Program. **45**, 503 (1989).

[39] https://lbfgspp.statr.me

[40] Seconds to minutes on Intel(R) Core(TM) i5-1145G7 @ 2.60 GHz.

[41] A. Condon and R. M. Karp, Algorithms for graph partitioning on the planted partition model, Random Struct. Alg. **18**, 116 (2001).

[42] B. Karrer and M. E. J. Newman, Stochastic blockmodels and community structure in networks, Phys. Rev. E **83**, 016107 (2011).

[43] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová, Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications, Phys. Rev. E **84**, 066106 (2011).

[44] H. Kesten and B. P. Stigum, A limit theorem for multidimensional Galton-Watson processes, Ann. Math. Stat. **37**, 1211 (1966).

[45] L. Gulikers, M. Lelarge, and L. Massoulié, An impossibility result for reconstruction in the degree-corrected stochastic block model, Ann. Appl. Probab. **28**, 3002 (2018).

[46] L. Peel, T. P. Peixoto, and M. De Domenico, Statistical inference links data and theory in network science, Nat. Commun. **13**, 6794 (2022).

[47] T. P. Peixoto, Bayesian stochastic blockmodeling, in *Advances in Network Clustering and Blockmodeling*, edited by P. Doreian, V. Batagelj, and A. Ferligoj (Wiley, New York, 2019), pp. 289–332.

[48] J. Rissanen, A universal prior for integers and estimation by minimum description length, Ann. Stat. **11**, 416 (1983).

[49] T. P. Peixoto, Nonparametric Bayesian inference of the microcanonical stochastic block model, Phys. Rev. E **95**, 012317 (2017).

[50] F. Giuffrida, T. Squartini, P. Grünwald, and D. Garlaschelli, Description length of canonical and microcanonical models, Phys. Rev. Res. **7**, 043057 (2025).

[51] W. W. Zachary, An information flow model for conflict and fission in small groups, J. Anthropol. Res. **33**, 452 (1977).

[52] L. A. Adamic and N. Glance, The political blogosphere and the 2004 US election: Divided they blog, in *Proceedings of the 3rd International Workshop on Link Discovery*, KDD'05 (ACM Press, New York, NY, 2005), pp. 36–43.

[53] J. B. Tenenbaum, V. de Silva, and J. C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science **290**, 2319 (2000).

[54] S. Emmons and P. J. Mucha, Map equation with metadata: Varying the role of attributes in community detection, Phys. Rev. E **100**, 022301 (2019).

[55] L. Peel, D. B. Larremore, and A. Clauset, The ground truth about metadata and community detection in networks, Sci. Adv. **3**, e1602548 (2017).

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

[57] M. Rosvall and C. T. Bergstrom, Maps of random walks on complex networks reveal community structure, Proc. Natl. Acad. Sci. USA **105**, 1118 (2008).

[58] J. P. A. Ioannidis, Why most published research findings are false, PLoS Med. **2**, e124 (2005).

[59] J. P. Simmons, L. D. Nelson, and U. Simonsohn, False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant, Psychol. Sci. **22**, 1359 (2011).

[60] O. J. Dunn, Multiple comparisons among means, J. Am. Stat. Assoc. **56**, 52 (1961).

[61] Y. Benjamini and Y. Hochberg, Controlling the false discovery rate: A practical and powerful approach to multiple testing, J. R. Stat. Soc. B **57**, 289 (1995).

[62] L. Anselin, An introduction to spatial data science with GeoDa, in *Exploring Spatial Data* (Chapman and Hall/CRC Press, Boca Raton, FL, 2024), Vol. 1.

[63] R. Arthur, Correlation and autocorrelation of data on complex networks, EPJ Data Sci. **14**, 6 (2025).

[64] B. Efron and T. Hastie, *Computer Age Statistical Inference, Student Edition: Algorithms, Evidence, and Data Science* (Cambridge University Press, Cambridge, 2021), Vol. 6.