Electronic Journal of Statistics
Vol. 10 (2016) 3153–3178
ISSN: 1935-7524
DOI: 10.1214/16-EJS1206

Community detection in networks with node features

Yuan Zhang

Department of Statistics, Ohio State University
Columbus, OH, 43210, USA
e-mail: yzhanghf@stat.osu.edu

Elizaveta Levina and Ji Zhu

Department of Statistics, University of Michigan
Ann Arbor, MI, 48109-1107, USA
e-mail: elevina@umich.edu; jizhu@umich.edu

Abstract: Many methods have been proposed for community detection in networks, but most of them do not take into account additional information on the nodes that is often available in practice. In this paper, we propose a new joint community detection criterion that uses both the network edge information and the node features to detect community structures. One advantage our method has over existing joint detection approaches is the flexibility of learning the impact of different features which may differ across communities. Another advantage is the flexibility of choosing the amount of influence the feature information has on communities. We show the method performs well on simulated and real networks.

Keywords and phrases: Network communities, node features, joint detection.

Received October 2015.

Contents

1. Introduction

Community detection is a fundamental problem in network analysis, extensively studied in a number of domains – see [20] and [21] for some examples of applications. A number of approaches to community detection are based on probabilistic models for networks with communities, such as the stochastic block model [10], the degree-corrected stochastic block model [12], and the latent factor model [9]. Other approaches work by optimizing a criterion measuring the strength of community structure in some sense, often through spectral approximations. Examples include normalized cuts [22], modularity [17, 16], and many variants of spectral clustering, e.g., [19].

Many of the existing methods detect communities based only on the network adjacency matrix. However, we often have additional information on the nodes (node features), and sometimes edges as well, for example, [26], [25] and [11]. In many networks the distribution of node features is correlated with community structure [15], and thus a natural question is whether we can improve community detection by using the node features. Several generative models for jointly modeling the edges and the features have been proposed, including the network random effects model [8], the embedding feature model [31], the latent variable model [6], the discriminative approach [30], the latent multi-group membership graph model [14], the social circles model for ego networks [15], the communities from edge structure and node attributes (CESNA) model [29], the Bayesian Graph Clustering (BAGC) model [28], the topical communities and personal interest (TCPI) model [7] and the modified stochastic block model [18]. The latter paper was written after this work was completed, and while its goals are somewhat similar to ours by also learning the relationship between the features and the network from data, it is very different in that it postulates a model connecting them in a particular way. Most of these models are designed for specific feature types, and their effectiveness depends heavily on the correctness of model specification. Model-free approaches include weighted combinations of the network and feature similarities [27, 2], attribute-structure mining [23], simulated annealing clustering [3], and compressive information flow [24]. Most methods in this category use all the features in the same way without determining which ones influence the community structure and which do not, and lack flexibility in how to balance the network information with the information coming from its node features, which do not always agree. Including irrelevant node features can only hurt community detection by adding in noise, while selecting features that by themselves cluster strongly may not correspond to features that correlate with the community structure present in the adjacency matrix.

In this paper, we propose a new joint community detection criterion that uses both the network adjacency matrix and the node features. The idea is that by properly weighing edges according to feature similarities on their end nodes, we strengthen the community structure in the network thus making it easier to detect. Rather than using all available features in the same way, we learn which features are most helpful in identifying the community structure from data. Intuitively, our method looks for an agreement between clusters suggested by two data sources, the adjacency matrix and the node features. Numerical experiments on simulated and real networks show that our method performs well compared to methods that use either the network alone or the features alone for clustering, as well as to a number of benchmark joint detection methods.

2. The joint community detection criterion

Our method is designed to look for assortative community structure, that is, the type of communities where nodes are more likely to connect to each other if they belong to the same community, and thus there are more edges within communities than between. This is a very common intuitive definition of communities which is incorporated in many community detection criteria, for example, modularity [16]. Our goal is to use such a community detection criterion based on the adjacency matrix alone, and add feature-based edge weights to improve detection. Several criteria using the adjacency matrix alone are available, but having a simple criterion linear in the adjacency matrix makes optimization much more feasible in our particular situation, and we propose a new criterion which turns out to work particularly well for our purposes. Let A denote the adjacency matrix with $A_{ij} = 0$ if there is no edge between nodes i and j, and otherwise $A_{ij} > 0$ which can be either 1 for unweighted networks or the edge weight for weighted networks. The community detection criterion we start from is a very simple analogue of modularity, to be maximized over all possible label assignments e:

$$R(e;\alpha) = \sum_{k=1}^{K} \frac{1}{|\mathcal{E}_k|^\alpha} \sum_{i,j\in\mathcal{E}_k} A_{ij} \ . \tag{2.1}$$

Here e is the vector of node labels, with $e_i = k$ if node i belongs to community k, for $k = 1,\dots,K$, $\mathcal{E}_k = \{i : e_i = k\}$, and $|\mathcal{E}_k|$ is the number of nodes in community k. We assume each node belongs to exactly one community, and the number of communities K is fixed and known. Rescaling by $|\mathcal{E}_k|^\alpha$ is designed to rule out trivial solutions that put all nodes in the same community, and $\alpha > 0$ is a tuning parameter. When $\alpha = 2$, the criterion is approximately the sum of edge densities within communities, and when $\alpha = 1$, the criterion is the sum of average "within community" degrees, which both intuitively represent community structure. Varying the tuning parameter α allows the user to penalize unbalanced communities (with larger α, see Sections 3.1 and A.2 for details), which, to the best of our knowledge, makes the criterion (2.1) the first method that allows such tuning. This criterion can be shown to be consistent under the stochastic block model, see Section 4.

The ideal use of features with this criterion would be to use them to up-weigh edges within communities and down-weigh edges between them, thus enhancing the community structure in the observed network and making it easier to detect. However, node features may not be perfectly correlated with community structure, different communities may be driven by different features, as pointed out by [15], and features themselves may be noisy. Thus we need to learn the impact of different features on communities as well as balance the roles of the network itself and its features. Let f_i denote the p-dimensional feature vector of node i. We propose a *joint community detection criterion* (JCDC),

$$R(e, \beta; w_n) = \sum_{k=1}^{K} \frac{1}{|\mathcal{E}_k|^\alpha} \sum_{i,j \in \mathcal{E}_k} A_{ij} W(f_i, f_j, \beta_k; w_n) \tag{2.2}$$

where α is a tuning parameter as in (2.1), $\beta_k \in \mathbb{R}^p$ is the coefficient vector that defines the impact of different features on the kth community, and $\beta := \{\beta_1, \dots, \beta_K\}$. The criterion is then maximized over both e and β. Having a different β_k for each k allows us to learn the roles different features may play in different communities. The balance between the information from A and $F := \{f_1, \dots, f_n\}$ is controlled by w_n, another tuning parameter which in general may depend on n.

For the sake of simplicity, we model the edge weight $W(f_i, f_j, \beta_k; w_n)$ as a function of the node features f_i and f_j via a p-dimensional vector of their similarity measures $\phi_{ij} = \phi(f_i, f_j)$. The choice of similarity measures in ϕ depends on the type of f_i (for example, on whether the features are numerical or categorical) and is determined on a case by case basis; the only important property is that ϕ assigns higher values to features that are more similar. Note that this trivially allows the inclusion of edge features as well as node features, as long as they are converted to some sort of similarity. To eliminate potential differences in units and scales, we standardize all ϕ_{ij} along each feature dimension. Finally, the function W should be increasing in $\langle \phi_{ij}, \beta \rangle$, which can be viewed as the "overall similarity" between nodes, and for optimization purposes it is convenient to take W to be concave. Here we use the exponential function,

$$w_{ijk} = W(f_i, f_j, \beta_k; w_n) = w_n - e^{-\langle \phi_{ij}, \beta_k \rangle} \tag{2.3}$$

One can use other functions of similar shapes, for example, the logit exponential function, which we found empirically to perform similarly.

3. Estimation

The joint community detection criterion needs to be optimized over both the community assignments e and the feature parameters β. Using block coordinate descent, we optimize JCDC by alternately optimizing over the labels with fixed parameters and over the parameters with fixed labels, and iterating until convergence.

3.1. Optimizing over label assignments with fixed weights

When parameters β are fixed, all edge weights w_{ijk}'s can be treated as known constants. It is infeasible to search over all n^K possible label assignments, and, like many other community detection methods, we rely on a greedy label switching algorithm to optimize over e, specifically, the tabu search [5], which updates the label of one node at a time. Since our criterion involves the number of nodes in each community $|\mathcal{E}_k|$, no easy spectral approximations are available. Fortunately, our method allows for a simple local approximate update which does not require recalculating the entire criterion. For a given node i considered for label switching, the algorithm will assign it to community k rather than l if

$$\frac{S_{kk} + 2S_{i\leftrightarrow k}}{(|\mathcal{E}_k| + 1)^\alpha} + \frac{S_{ll}}{|\mathcal{E}_l|^\alpha} > \frac{S_{kk}}{|\mathcal{E}_k|^\alpha} + \frac{S_{ll} + 2S_{i\leftrightarrow l}}{(|\mathcal{E}_l| + 1)^\alpha} , \tag{3.1}$$

where S_{kk} is twice the total edge weights in community k, and $S_{i\leftrightarrow k}$ is the sum of edge weights between node i and all the nodes in $\mathcal{E}_k$. When $|\mathcal{E}_k|$ and $|\mathcal{E}_l|$ are large, we can ignore $+1$ in the denominators, and (3.1) becomes

$$\frac{S_{i\leftrightarrow k}}{|\mathcal{E}_k|} \cdot \frac{|\mathcal{E}_k|^{1-\alpha}}{|\mathcal{E}_l|^{1-\alpha}} > \frac{S_{i\leftrightarrow l}}{|\mathcal{E}_l|} , \tag{3.2}$$

which allows for a "local" update for the label of node i without calculating the entire criterion. This also highlights the impact of the tuning parameter α: when $\alpha = 1$, the two sides of (3.2) can be viewed as averaged weights of all edges connecting node i to communities $\mathcal{E}_k$ and $\mathcal{E}_l$, respectively. Then our method assigns node i to the community with which it has the strongest connection. When $\alpha \neq 1$, the left hand side of (3.2) is multiplied by a factor $(|\mathcal{E}_k|/|\mathcal{E}_l|)^{1-\alpha}$. Suppose $|\mathcal{E}_k|$ is larger than $|\mathcal{E}_l|$; then choosing $0 < \alpha < 1$ indicates a preference for assigning a node to the larger community, while $\alpha > 1$ favors smaller communities. A detailed numerical investigation of the role of α is provided in Section A.2.

The edge weights involved in (3.2) depend on the tuning parameter w_n. When $\beta = 0$, all weights are equal to $w_n - 1$. On the other hand, $w_{ijk} \leq w_n$ for all values of β. Therefore, $w_n/(w_n - 1)$ is the maximum amount by which our method can reweigh an edge. When w_n is large, $w_n/(w_n - 1) \approx 1$, and thus the information from the network structure dominates. When w_n is close to 1, the ratio is large and the feature-driven edge weights have a large impact. See Section A.2 for more details on the choice of w_n.

While the tuning parameter w_n controls the amount of influence features can have on community detection, it does not affect the estimated parameters β for a fixed community assignment. This is easy to see from rearranging terms in (2.2):

$$R(e, \beta; w_n) = w_n \sum_{k=1}^{K} \frac{1}{|\mathcal{E}_k|^\alpha} \sum_{i,j \in \mathcal{E}_k} A_{ij} - g(e, A, \beta, \phi) \tag{3.3}$$

where the function g does not depend on w_n. Note that the term containing w_n does not depend on β.

3.2. Optimizing over weights with fixed label assignments

Since we chose a concave edge weight function (2.3), for a given community assignment e the joint criterion is a concave function of β_k, and it is straightforward to optimize over β_k by gradient ascent. The role of β_k is to control the impact of different features on each community. One can show by a Taylor-series type expansion around the maximum (details omitted) and also observe empirically that for our method, the estimated $\hat{\beta}_k$'s are correlated with the feature similarities between nodes in community k. In other words, our method tends to produce a large estimated $\hat{\beta}_k^{(\ell)}$ for a feature with high similarity values $\phi_{ij}^{(\ell)}$'s for $i, j \in \mathcal{E}_k$. However, in the extreme case, the optimal $\hat{\beta}_k^{(\ell)}$ can be $+\infty$ if all $\phi_{ij}^{(\ell)}$'s are positive in community k or $-\infty$ if all $\phi_{ij}^{(\ell)}$'s are negative (recall that similarities are standardized, so this cannot happen in all communities). To avoid these extreme solutions, we subtract a penalty term $\lambda\|\beta\|_1$ from the criterion (2.2) while optimizing over β. We use a very small value of λ ($\lambda = 10^{-5}$ everywhere in the paper) which safeguards against numerically unstable solutions but has very little effect on other estimated coefficients.

The complete algorithm is given as pseudo-code in Section A.1. The computational cost is heaviest for the step of updating community assignments; some numerical studies of the computational cost for varying network sizes and numbers of communities are reported in Section A.4.

4. Consistency

In this section, we focus on the properties of community detection criterion (2.1), which serves as the foundation for the joint criterion (2.2). We show that under certain models, (2.1) is asymptotically consistent for community detection.

Let $\mathbb{P}(A_{ij} = 1) = \rho_n P_{c_i c_j}$ where ρ_n is a factor controlling the overall edge density and $c = (c_1, \dots, c_n)$ is the vector of true labels. Assume the following regularity conditions hold:

1. Let $\mathcal{C}_k := \{i : c_i = k\}$. There exists a global constant π_0 such that $|C_k| \geq \pi_0 n > 0$ for all k.
2. For all $1 \leq k < l \leq K$, $2(K-1)P_{kl} < \min(P_{kk}, P_{ll})$.

Condition 1 guarantees that the proportions of nodes in each community do not vanish asymptotically. Condition 2 enforces assortativity. Note that Condition 2 can be substantially relaxed if we use $\alpha = 1$, as we do for all the numerical results later in the paper. In that case, we only need $P_{kl} < \min(P_{kk}, P_{ll})$ for all $1 \leq k < l \leq K$.

Since the estimated labels e are only defined up to an arbitrary permutation of communities, we measure the agreement between e and c by $|e - c| = \min_{\sigma \in \mathcal{P}_K} \frac{1}{n} \sum_{i=1}^n \mathbf{1}(\sigma(e_i) \neq c_i)$, where $\mathcal{P}_K$ is the set of all permutations of $\{1, \dots, K\}$.

Recall we assume that the fraction of nodes in each community is bounded below by some π_0, thus we restrict the range of the estimated community size to be at least $\pi_0 n$. Denote $\mathcal{E}^{\pi_0} = \{(\mathcal{E}_1, \ldots, \mathcal{E}_K) : \min_k |\mathcal{E}_k| \geq \pi_0 n\}$, and let

$$\hat{e} = \arg\max_{e \in \mathcal{E}^{\pi_0}} R(e; \alpha) \ ,$$

where $R(e; \alpha)$ is the criterion defined in (2.1), and $\hat{e}$ is defined up to a permutation of community labels. We suppress the dependence of $\hat{e}$ on α wherever there is no confusion.

Theorem 1. *Under conditions 1 and 2, if $n\rho_n \to \infty$, and the parameter α satisfies*

$$\frac{\max_{k,l} 2(K-1)P_{kl}}{\min_{k,l}(P_{kk}, P_{ll})} \leq \alpha \leq 1 \tag{4.1}$$

then we have, for any fixed $\delta > 0$,

$$\mathbb{P}\left(|\arg\max_{e \in \mathcal{E}^{\pi_0}} R(e; \alpha) - c| > \delta \right) \to 0 \tag{4.2}$$

The proof is given in Section A.5.

The natural question to ask next is whether the joint criterion is also consistent, and under what conditions. This brings to light the fundamentally different asymptotic behavior of community detection and multivariate clustering. Theorem 1 indicates that, under certain conditions, (2.1) can obtain consistent community detection from the adjacency matrix alone, with all nodes labeled correctly with high probability. On the other hand, suppose the node features are generated from a mixture of multivariate Gaussians, a standard set-up used for analysis of clustering. Then it is well known that the only possible way to obtain consistent clustering from a Gaussian mixture is to have the cluster variances go to 0, which makes the problem trivial; otherwise the best one can hope to achieve is the Bayes risk for the corresponding classification problem. Thus asymptotic analysis of the joint criterion can only tell us that the adjacency matrix alone is enough for consistency when there is enough signal in the network, and in that situation consistency of JCDC can be maintained by diminishing the influence of node features as the size of the network grows. To see the real benefits of including node features, we have to look at the finite sample performance, and there are not enough analytic tools available right now to carry out an analysis of this nature. Nonetheless, we see marked improvements empirically with inclusion of node features, as was also reported by [2].

5. Simulation studies

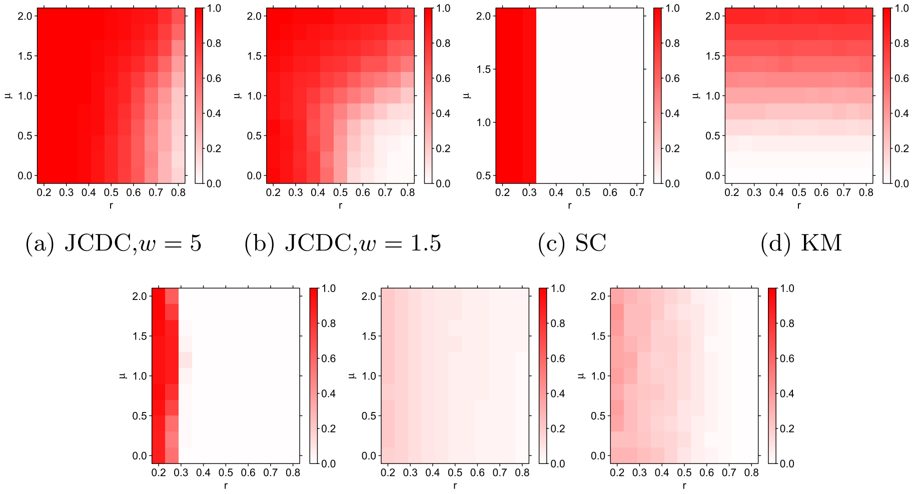
We compare JCDC to three representative benchmark methods which use both the adjacency matrix and the node features: CASC (Covariate Assisted Spectral Clustering, [2]), CESNA (Communities from Edge Structure and Node Attributes, [29]), and BAGC (BAyesian Graph Clustering, [28]). In addition, we also include two standard methods that use either the network adjacency alone (SC, spectral clustering on the Laplacian regularized with a small constant

$\tau = 1e{-}7$, as in [1]), or the node features alone (KM, K-means performed on the p-dimensional node feature vectors, with 10 random initial starting values). We also considered three different initialization methods for JCDC: random starting values, spectral clustering regularized with the same small constant, and the pseudo-likelihood method [1]. Detailed comparisons are provided in Section A.3. We found that initializing with spectral clustering produces the best results, and thus we initialize JCDC with spectral clustering for the rest of the paper.

To test these methods, we generate networks with $n = 150$ nodes and $K = 2$ communities of sizes 100 and 50 from the degree-corrected stochastic block model as follows. The edges are generated independently with probability $\theta_i\theta_j p$ if nodes i and j are in the same community, and $r\theta_i\theta_j p$ if nodes i and j are in different communities. We set $p = 0.1$ and vary r from 0.25 to 0.75. We set 5% of the nodes in each community to be "hub" nodes with the degree correction parameter $\theta_i = 10$, and for the remaining nodes set $\theta_i = 1$. All resulting products are thresholded at 0.99 to ensure there are no probability values over 1. These settings result in the average expected node degree ranging approximately from 22 to 29.

For each node i, we generate $p = 2$ features, with one "signal" feature related to the community structure and one "noise" feature whose distribution is the same for all nodes. The "signal" feature follows the distribution $N(\mu, 1)$ for nodes in community 1 and $N(-\mu, 1)$ for nodes in community 2, with μ varying from 0.5 to 2 (larger μ corresponds to stronger signal). For use with CESNA, which only allows categorical node features, we discretize the continuous node features by partitioning the real line into 20 bins using the $0.05, 0.1, \ldots, 0.95$-th quantiles. For the JCDC, based on the study of the tuning parameters in Section A.2, we use $\alpha = 1$ and compare two values of w_n, $w_n = 1.5$ and $w_n = 5$. Finally, agreement between the estimated communities and the true community labels is measured by normalized mutual information, a measure commonly used in the network literature which ranges between 0 (random guessing) and 1 (perfect agreement). For each configuration, we repeat the experiments 30 times, and record the average NMI over 30 replications.

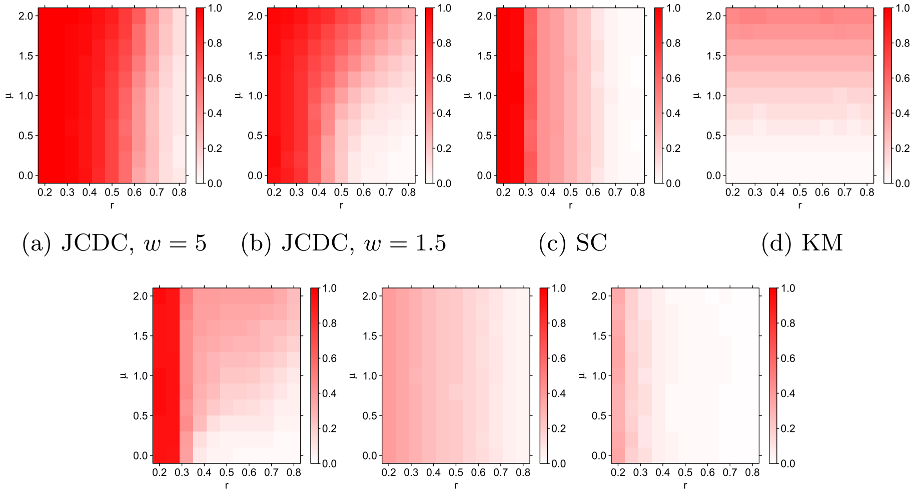
Figure 1 shows the heatmaps of average NMI for all methods under these settings, as a function of r and μ. As one would expect, the performance of spectral clustering (c), which uses only the network information, is only affected by r (the larger r is, the harder the problem), and the performance of K-means (d), which uses only the features, is only affected by μ (the larger μ is, the easier the problem). JCDC is able to take advantage of both network and feature information by estimating the coefficients β from data, and its performance only deteriorates when neither is informative. The informative features are more helpful with a larger value of w (a), and conversely uninformative features affect performance slightly more with a lower value of w (b), but this effect is not strong. CASC (e) appears to inherit the sharp phase transition from spectral clustering, which forms the basis of CASC; the sharp transition is perhaps due to different community sizes and hub nodes, which are both challenging to spectral clustering. CESNA (f) and BAGC (g) do not perform as well overall, with BAGC often clustering all the hub nodes into one community.



(a) JCDC,$w = 5$ (b) JCDC,$w = 1.5$ (c) SC (d) KM

(e) CASC (f) CESNA (g) BAGC

Fig 1. *Performance of different methods measured by normalized mutual information as a function of r (out-in probability ratio) and μ (feature signal strength), for networks with $K = 2$, $n_1 = 100$, $n_2 = 50$.*

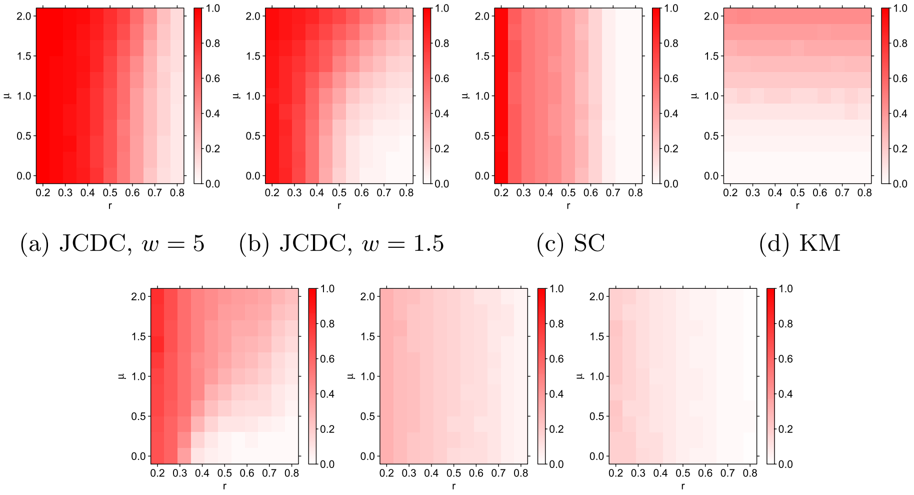


(a) JCDC, $w = 5$ (b) JCDC, $w = 1.5$ (c) SC (d) KM

(e) CASC (f) CESNA (g) BAGC

Fig 2. *Performance of different methods measured by normalized mutual information as a function of r (out-in probability ratio) and μ (feature signal strength), for networks with $K = 3$ and $n_1 = n_2 = n_3 = 50$.*

Figures 2 and 3 show the NMI results for all methods on networks with $K = 3$ communities and $n = 150$ nodes, Figure 2 for the case of communities of equal sizes, and Figure 3 for the unbalanced case. The setting for generating the



(a) JCDC, $w = 5$ (b) JCDC, $w = 1.5$ (c) SC (d) KM

(e) CASC (f) CESNA (g) BAGC

FIG 3. *Performance of different methods measured by normalized mutual information as a function of r (out-in probability ratio) and μ (feature signal strength), for networks with $K = 3$ and $n_1 = 30$, $n_2 = 50$, $n_3 = 70$.*

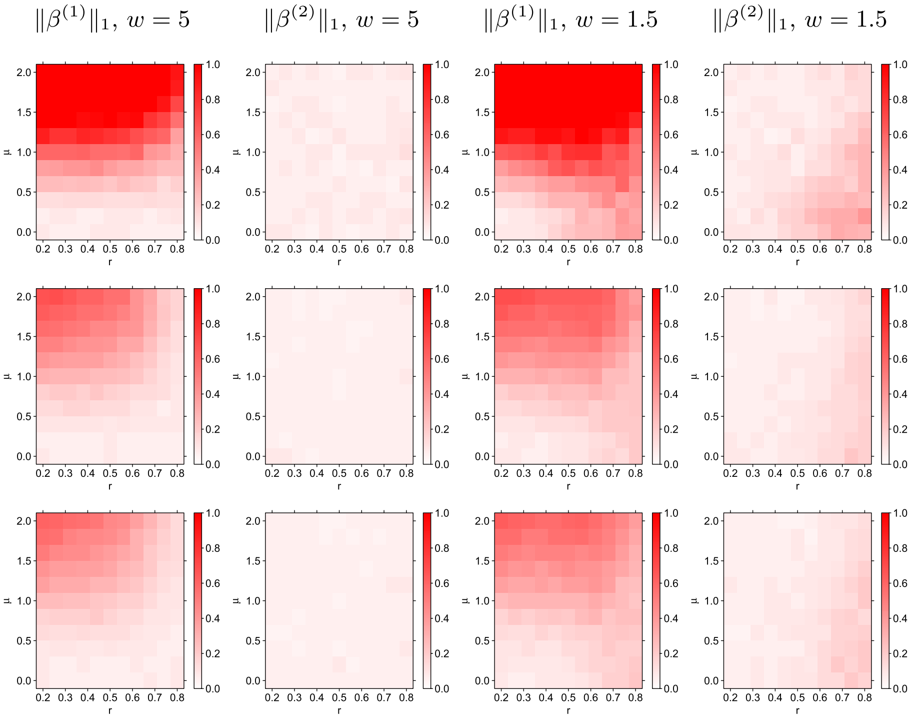
adjacency matrix is the same as in the previous simulation, and for node features, we set the distribution of the "signal" feature to be $N(\mu, 1)$ for community 1, $N(0, 1)$ for community 2 and $N(-\mu, 1)$ for community 3. The results are similar to those in Figure 1, and the JCDC method has an advantage over other benchmark methods for most (r, μ) combinations.

Finally, we inspect the estimated feature coefficients $\hat{\beta}_k$ in each community. Ideally, we should obtain larger estimated values of β_k^ℓ when the ℓ-th feature is helpful in identifying nodes in community k. Since in the simulation settings the features affect all three communities in the same way, we report $\|\beta^{(1)}\|_1/K$ and $\|\beta^{(2)}\|_1/K$ for the two features, averaged across communities, in Figure 4. Overall, the coefficients for the "signal" feature are substantially larger than the estimated coefficients for the "noise" feature. Further, as μ decreases and the feature signal gets weaker, the estimated coefficients for the signal feature also get smaller, while they are not affected much by changes in the network parameter r.

6. Data applications

6.1. *The world trade network*

The world trade network [4] connects 80 countries based on the amount of trade of metal manufactures between them in 1994, or when not available for that year, in 1993 or 1995. Nodes are countries and edges represent positive



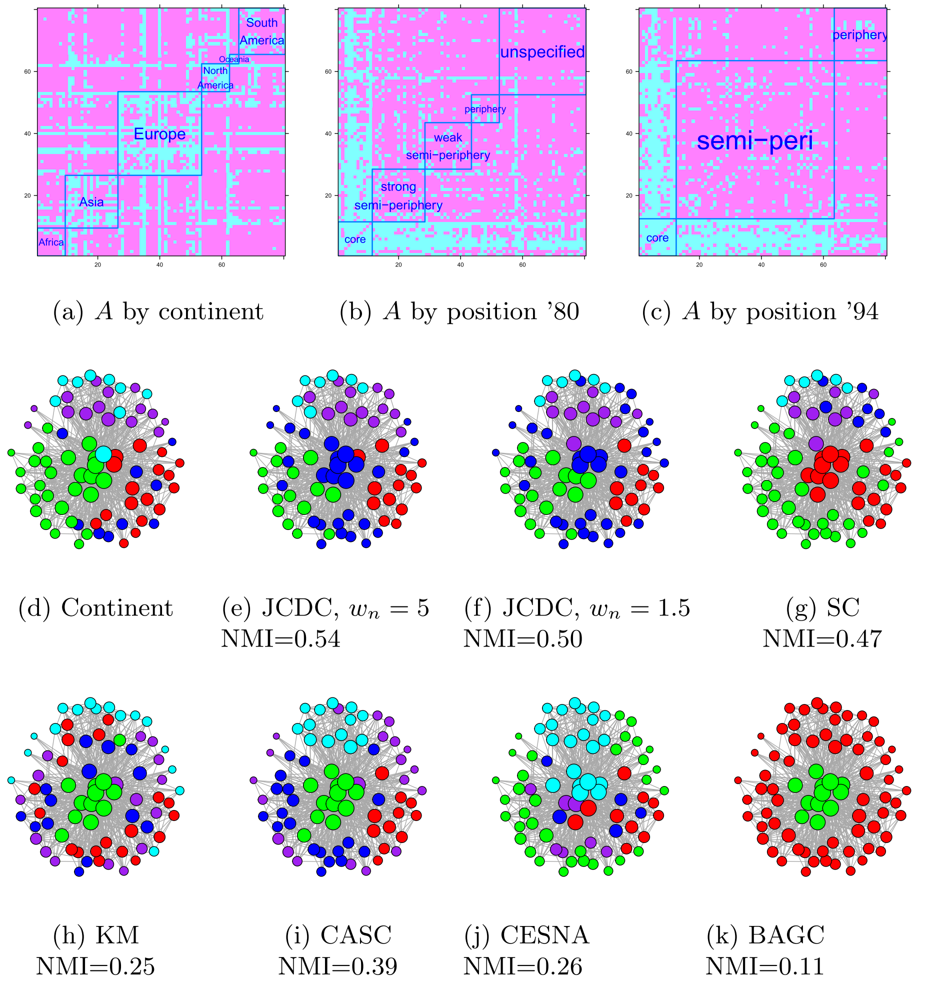
$\|\beta^{(1)}\|_1$, $w = 5$ $\|\beta^{(2)}\|_1$, $w = 5$ $\|\beta^{(1)}\|_1$, $w = 1.5$ $\|\beta^{(2)}\|_1$, $w = 1.5$

FIG 4. *Estimated $\|\hat{\beta}^{(\ell)}\|_1/K$ values. First row: $K = 2$, $n_1 = 100$, $n_2 = 50$; second row: $K = 3$, $n_1 = n_2 = n_3 = 50$; third row: $K = 3$, $n_1 = 30$, $n_2 = 50$, $n_3 = 70$.*

amount of import and/or export between the countries. Each country also has three categorical features: the continent (Africa, Asia, Europe, N. America, S. America, and Oceania), the country's structural position in the world system in 1980 (core, strong semi-periphery, weak semi-periphery, periphery) and in 1994 (core, semi-periphery, periphery). Figures 5 (a) to (c) show the adjacency matrix rearranged by sorting the nodes by each of the features. The partition by continent (Figure 5(a)) clearly shows community structure, whereas the other two features show hubs (core status countries trade with everyone), and no assortative community structure. We will thus compare partitions found by all the competing methods to the continents, and omit the three Oceania countries from further analysis because no method is likely to detect such a small community. The two world position variables ('80 and '94) will be used as features, treated as ordinal variables.

The results for all methods are shown in Figure 5, along with NMI values comparing the detected partition to the continents. All methods were run with the true value $K = 5$.

The result of spectral clustering agrees much better with the continents than that of K-means, indicating that the community structure in the adjacency matrix is closer to the continents than the structure contained in the node features.



(a) A by continent (b) A by position '80 (c) A by position '94

(d) Continent (e) JCDC, $w_n = 5$ NMI=0.54 (f) JCDC, $w_n = 1.5$ NMI=0.50 (g) SC NMI=0.47

(h) KM NMI=0.25 (i) CASC NMI=0.39 (j) CESNA NMI=0.26 (k) BAGC NMI=0.11

FIG 5. *(a)-(c): the adjacency matrix ordered by different node features; (d) network with nodes colored by continent (taken as ground truth); blue is Africa, red is Asia, green is Europe, cyan is N. America and purple is S. America. (e)-(k) community detection results from different methods; colors are mated to (d) in the best way possible.*

JCDC obtains the highest NMI value, CASC performs similarly to spectral clustering, whereas CESNA and BAGC both fail to recover the continent partition. Note that no method was able to estimate Africa well, likely due to the disassortative nature of its trade seen in Figure 5 (a). Figure 5 (e) indicates that JCDC estimated N. America, S. America and Asia with high accuracy, but split Europe into two communities, since it was run with $K = 5$ and could not pick up Africa due to its disassortative structure. Table 1 contains the estimated feature coefficients, suggesting that in 1980 the "world position" had the most influence on the connections formed by Asian countries, whereas in 1994 world position mattered most in Europe.

Table 1
Feature coefficients $\hat{\beta}_k$ estimated by JCDC with $w = 5$. Best match is determined by majority vote.

Community	Best match	Position '80	Position '94
blue	Europe	0.000	0.143
red	Asia	0.314	0.127
green	Europe	0.017	0.204
cyan	N. America	0.107	0.000
purple	S. America	0.121	0.000

6.2. The lawyer friendship network

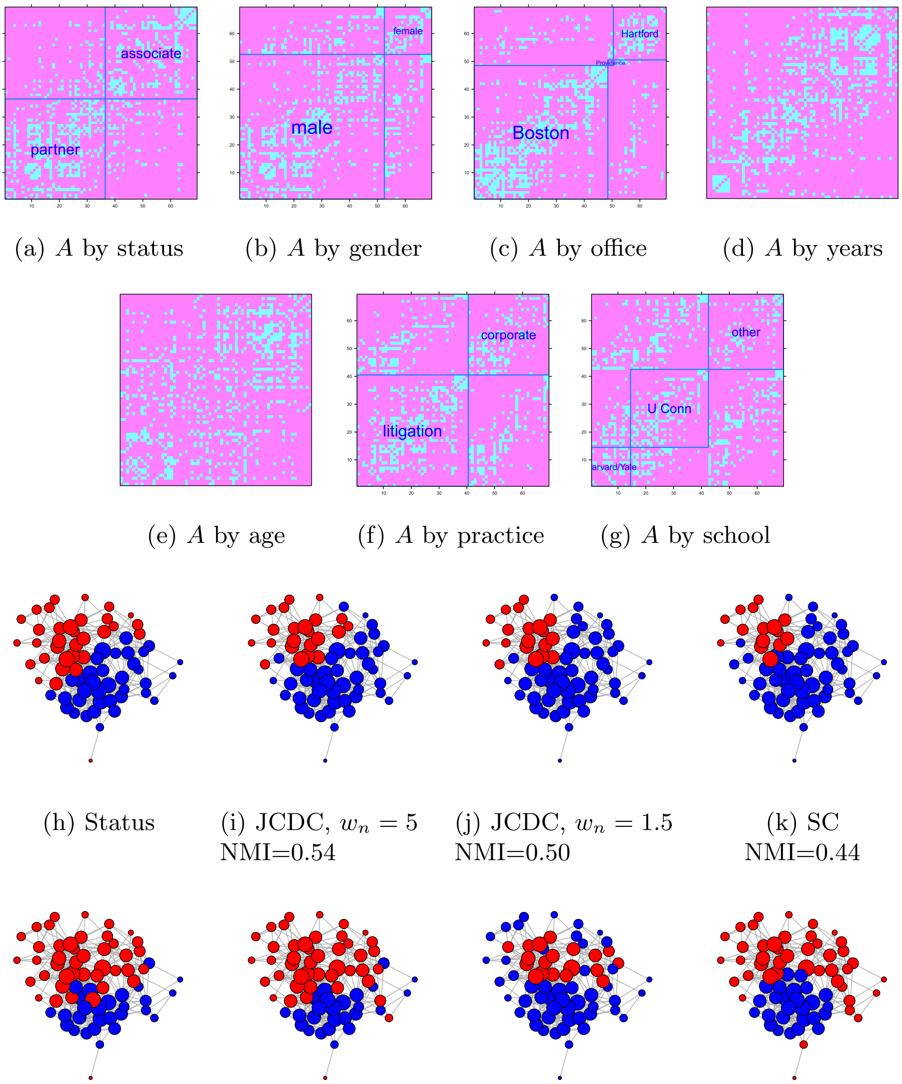
The second dataset we consider is a friendship network of 71 lawyers in a New England corporate law firm [13]. Seven node features are available: status (partner or associate), gender, office location (Boston, Hartford, or Providence, a very small office with only two non-isolated nodes), years with the firm, age, practice (litigation or corporate) and law school attended (Harvard, Yale, University of Connecticut, or other). Categorical features with M levels are represented by $M - 1$ dummy indicator variables. Figures 6 (a)-(g) show heatmap plots of the adjacency matrix with nodes sorted by each feature, after eliminating six isolated nodes. Partition by status (Figure 6(a)) shows a strong assortative structure, and so does partition by office (Figure 6(c)) restricted to Boston and Hartford, but the small Providence office does not have any kind of structure. Thus we chose the status partition as a reference point for comparisons, though other partitions are certainly also meaningful.

Communities estimated by different methods are shown in Figure 6 (i)-(o), all run with $K = 2$. Spectral clustering and K-means have equal and reasonably high NMI values, indicating that both the adjacency matrix and node features contain community information. JCDC obtains the highest NMI value, with $w_n = 5$ performing slightly better than $w_n = 1.5$. CASC improves upon spectral clustering by using the feature information, with NMI just slightly lower than that of JCDC with $w_n = 1.5$. CESNA and BAGC have much lower NMI values, possibly because of hub nodes, or because they detect communities corresponding to something other than status.

The estimated feature coefficients are shown in Table 2. Office location, years with the firm, and age appear to be the features most correlated with the community structure of status, for both partners and associates, which is natural. Practice, school, and gender are less important, though it may be hard to estimate the influence of gender accurately since there are relatively few women in the sample.

Table 2
Feature coefficients $\hat{\beta}_k$, JCDC with $w_n = 5$.

Comm.	gender	office	years	age	practice	school
partner	0.290	0.532	0.212	0.390	0.095	0.000
associate	0.012	0.378	0.725	0.320	0.118	0.097



(a) A by status (b) A by gender (c) A by office (d) A by years

(e) A by age (f) A by practice (g) A by school

(h) Status (i) JCDC, $w_n = 5$ NMI=0.54 (j) JCDC, $w_n = 1.5$ NMI=0.50 (k) SC NMI=0.44

(l) KM NMI=0.44 (m) CASC NMI=0.49 (n) CESNA NMI=0.07 (o) BAGC NMI=0.20

Fig 6. *(a)-(g): adjacency matrix with nodes sorted by features; (h): network with nodes colored by status (blue is partner, red is associate); (i)-(n): community detection results from different methods.*

7. Discussion

Our method incorporates feature-based weights into a community detection criterion, improving detection compared to using just the adjacency matrix or the node features alone, if the cluster structure in the features is related to the community structure in the adjacency matrix. It has the ability to estimate coefficients for each feature within each community and thus learn which features are correlated with the community structure. This ability guards against including noise features which can mislead community detection. The community detection criterion we use is designed for assortative community structure, with more connections within communities than between, and benefits the most from using features that have a similar clustering structure.

This work can be extended in several directions. It would be useful to develop fast (possibly approximate) algorithms to optimize (2.2). Variation in node degrees, often modeled via the degree-corrected stochastic block model [12] which regards degrees as independent of community structure, may in some cases be correlated with node features, and accounting for degree variation jointly with features can potentially further improve detection. Another useful extension is to overlapping communities. One possible way to do that is to optimize each summand in JCDC (2.2) separately and in parallel, which can create overlaps, but would require careful initialization. Statistical models that specify exactly how features are related to community assignments and edge probabilities can also be useful, though empirically we found no such standard models that could compete with the non-model-based JCDC on real data. This suggests that more involved and perhaps data-specific modeling will be necessary to accurately describe real networks, and some of the techniques we proposed, such as community-specific feature coefficients, could be useful in that context.

Appendix

A.1. The algorithm

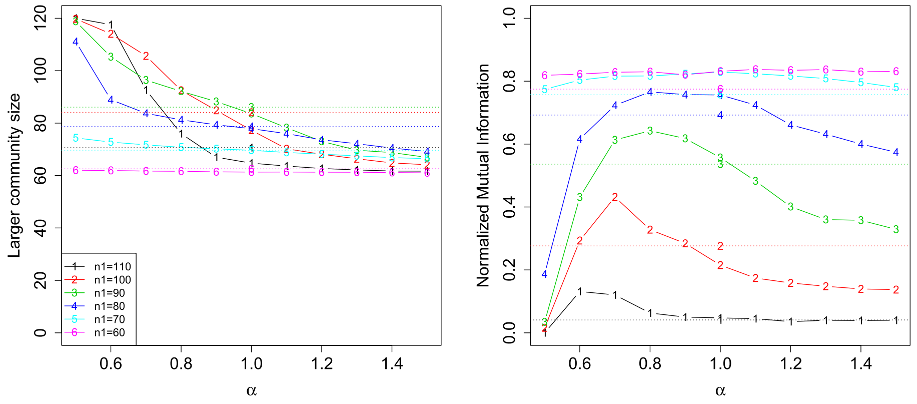
Algorithm 1 JCDC algorithm

```
Input: $A \in \mathbb{R}^{n\times n}$, $\phi \in \mathbb{R}^{n\times n\times p}$, $\alpha$, $w_n$, $\lambda$, $m$, $m_u$, $m_v$
for $t = 1$ to $m$ do
    for $u = 1$ to $m_u$ do
        for $i = 1$ to $n$ do Update:
            $i \leftarrow \arg\max_k \frac{S_{i\leftrightarrow k}}{|\mathcal{E}_k|^\alpha}$
    for $v = 1$ to $m_v$ do
        for $k = 1$ to $K$ do Update:
            $\beta_k \leftarrow \arg\max_{\beta_k} \frac{1}{|\mathcal{E}_k|^\alpha} \sum_{i,j\in\mathcal{E}_k} A_{ij} \left(w_n - e^{-\langle \phi_{ij}, \beta_k \rangle}\right) - \lambda \|\beta_k\|_1$
```

A.2. Choice of tuning parameters

The JCDC method involves two user-specified tuning parameters, α and w_n. In this section, we investigate the impact of these tuning parameters on community detection results via numerical experiments.

First we study the impact of α, which determines the algorithm's preference for larger or smaller communities. We study its effect on the estimated community size as well as on the accuracy of estimated community labels. We generate data from a stochastic block model with $n = 120$ nodes and $K = 2$ communities of sizes n_1 and $n_2 = n - n_1$. We set the within-community edge probabilities to 0.3 and between-community edge probabilities to 0.15, and vary n_1 from 60 to 110. Since α is not related to feature weights, we set features to a constant, resulting in unweighted networks. The results are averaged over 50 replications and shown in Figure 7.



(a) Size of the larger estimated community (b) Community detection accuracy

Fig 7. *(a) The size of the larger estimated community as a function of the tuning parameter α. (b) Estimation accuracy measured by NMI as a function of the tuning parameter α. Solid lines correspond to JCDC and horizontal dotted lines correspond to spectral clustering, which does not depend on α.*

We report the size of the larger estimated community in Figure 7(a), and the accuracy of community detection as measured by normalized mutual information (NMI) in Figure 7(b). For comparison, we also record the results from spectral clustering (horizontal lines in Figure 7), which do not depend on α. When communities are balanced ($n_1 = n_2 = 60$), JCDC performs well for all values of α, producing balanced communities and uniformly outperforming spectral clustering in terms of NMI. In general, larger values of α in JCDC result in more balanced communities, while smaller α's tend to produce a large and a small community. In terms of community detection accuracy, Figure 7(b) shows that the JCDC method outperforms spectral clustering over a range of values of α, and this range depends on how unbalanced the communities are. For simplicity and ease of interpretation, we set $\alpha = 1$ for all the simulations and data analysis reported in the main manuscript; however, it can be changed by the user if information about community sizes is available.

Next, we investigate the impact of w_n, which controls the influence of features. To study the trade-off between the two sources of information (network and features), we generate two different community partitions. Specifically, we consider two communities of sizes n_1 and n_2, with $n_1 + n_2 = n = 120$. We generate two label vectors c^A and c^F, with $c_i^A = 1$ for $i = 1, \ldots, n_1$ and $c_i^A = 2$ for $i = n_1 + 1, \ldots, n$, while the other label vector has $c_i^F = 1$ for $i = 1, \ldots, n_2$ and $c_i^F = 2$ for $i = n_2 + 1, \ldots, n$. Then the edges are generated from the stochastic block model based on c^A, and the node features are generated based on c^F. We generate two node features: one feature is sampled from the distribution $N(\mu, 1)$ if $c_i^F = 1$ and $N(0, 1)$ if $c_i^F = 2$; the other feature is sampled from $N(0, 1)$ if $c_i^F = 1$ and $N(-\mu, 1)$ if $c_i^F = 2$. We fix $\mu = 3$ and set $\alpha = 1$, as discussed above. We set the within- and between-community edge probabilities to 0.3 and 0.15, respectively, same as in the previous simulation, and vary the value of w_n from 1.1 to 10. Finally, we look at the the agreement between the estimated communities $\hat{e}$ and c_A and c_F, as measured by normalized mutual information. The results are shown in Figure 8.

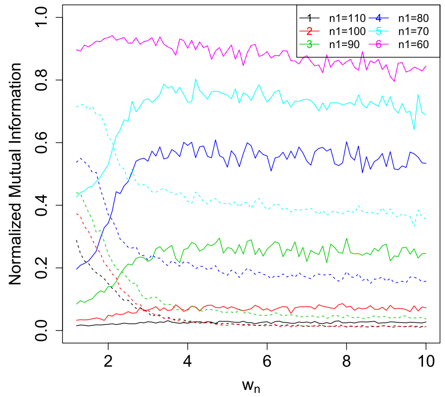
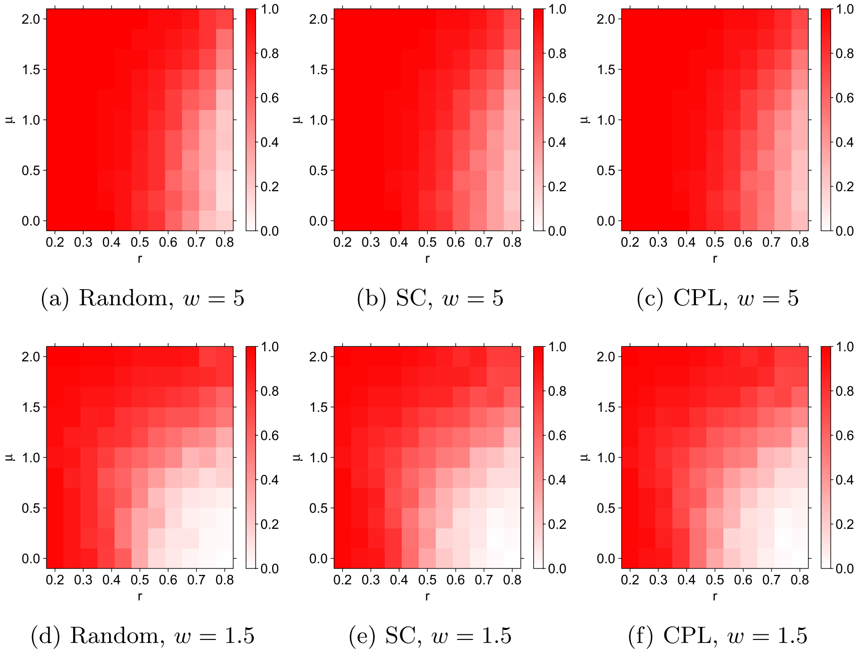


FIG 8. *MNI between the estimated community structure $\hat{e}$ and the network community structure c_A (solid lines) and the feature community structure c_F (dotted lines). Note that when $n_1 = n_2 = 60$, $c^A = c^F$, so the solid and dotted lines coincide.*

As we expect, smaller values of w_n give more influence to features and thus the estimated community structure agrees better with c^F than with c^A. As w_n increases, the estimated $\hat{e}$ becomes closer to c^A. In the manuscript, we compare two values of w_n, 1.5 and 5.

A.3. Initialization methods

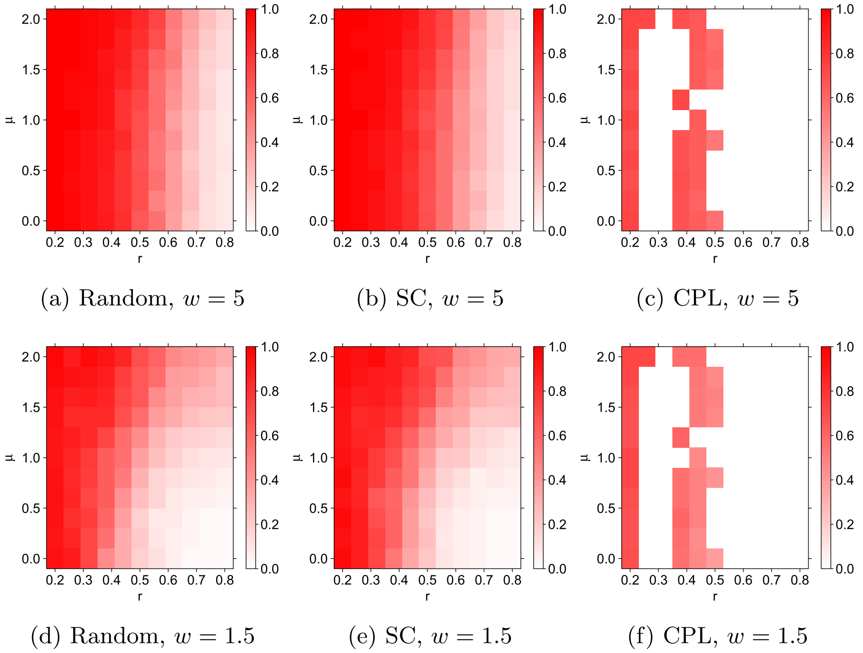
In this section, we compare three initialization methods for JCDC: random starts, spectral clustering regularized by a small constant $\tau = 1e - 7$, as in [1] (SC), and conditional pseudo-likelihood method (CPL), also from [1]. The results are shown in Figures 9, 10 and 11, for the three different simulation settings. Overall, spectral clustering performs the best. When $K = 3$ (Figures 10 and 11), CPL had some convergence issues, but when it does converge, the



(a) Random, $w = 5$ (b) SC, $w = 5$ (c) CPL, $w = 5$

(d) Random, $w = 1.5$ (e) SC, $w = 1.5$ (f) CPL, $w = 1.5$

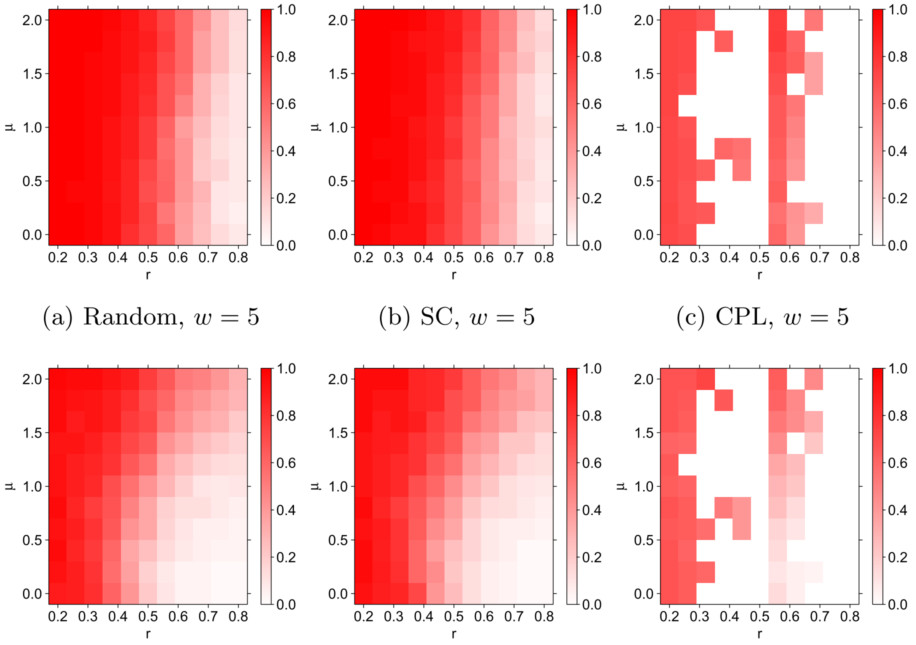
Fig 9. *NMI for JCDC using different initialization methods, $K = 2$ communities of sizes* $100, 50$.



(a) Random, $w = 5$ (b) SC, $w = 5$ (c) CPL, $w = 5$

(d) Random, $w = 1.5$ (e) SC, $w = 1.5$ (f) CPL, $w = 1.5$

Fig 10. *NMI for JCDC using different initialization methods, $K = 3$ communities of sizes* $30, 50, 70$.



(a) Random, $w = 5$ (b) SC, $w = 5$ (c) CPL, $w = 5$

(d) Random, $w = 1.5$ (e) SC, $w = 1.5$ (f) CPL, $w = 1.5$

FIG 11. *NMI for JCDC using different initialization methods, $K = 3$ communities of sizes* $50, 50, 50$.

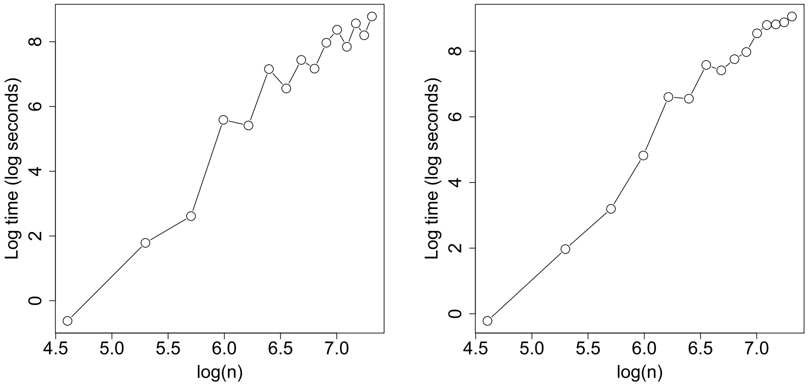
results initialized by CPL are similar to those initialized by SC. Random starts are only slightly worse that initializing by spectral clustering, suggesting that JCDC is not particularly sensitive to starting values.

A.4. Computational cost

The proposed algorithm uses a local greedy search to update estimated community labels and is thus computationally demanding. Empirically, we found that the proposed algorithm is able to estimate networks of $n = 500$ nodes within a few minutes. Networks with more than 1000 nodes, however, are challenging for the proposed algorithm. In Figure 12, we show the computational time of the proposed algorithm under varying network sizes. As we can see, as the network size increases, the computational time increases polynomially.

A.5. Proofs

We start with summarizing notation. Let $\mathcal{E}_1, \dots, \mathcal{E}_K$ be the estimated communities corresponding to the label vector e, and $\mathcal{C}_1, \dots, \mathcal{C}_K$ the true communities



(a) $K = 2$

(b) $K = 3$

Fig 12. *Running time of the proposed algorithm, in log(seconds) as a function of log(network size), where the network size ranges from 100 to 1500. For each* (n, K)*, the experiment was repeated 10 times and the average was plotted.*

corresponding to the label vector c. Define $R^0(e)$, the "population version" of R, as

$$R^0(e;\alpha) = \sum_{k=1}^{K} \frac{1}{|\mathcal{E}_k|^\alpha} \sum_{i,j\in\mathcal{E}_k} \rho_n P_{c_i c_j} \ .$$

Lemma 2. *If* $\rho_n n \to \infty$*, then*

$$\mathbb{P}\left(\max_{e\in\mathcal{E}^{\pi_0}} \frac{|R(e;\alpha) - R^0(e;\alpha)|}{\rho_n n^{2-\alpha}} > C \right) \to 0$$

for any global constant $C > 0$*.*

Proof of Lemma 2. By Bernstein's inequality,

$$\begin{aligned} \mathbb{P}\left(\left| \sum_{i,j\in\mathcal{E}_k} \left(A_{ij} - \rho_n P_{c_i c_j}\right) \right| \geq t \right) &\leq 2\exp\left(-\frac{\frac{1}{2}t^2}{\sum_{i,j\in\mathcal{E}_k} \mathrm{Var}(A_{ij}) + \frac{1}{3}t} \right) \\ &\leq 2\exp\left(-\frac{\frac{1}{2}t^2}{\rho_n n^2 + \frac{1}{3}t} \right) \end{aligned}$$

Setting $t = n^2\epsilon$, we have

$$\mathbb{P}\left(\left| \sum_{i,j\in\mathcal{E}_k} \left(A_{ij} - \rho_n P_{c_i c_j}\right) \right| \geq n^2\epsilon \right) \leq 2\exp\left(-\frac{\frac{1}{2}n^2\epsilon^2}{\rho_n + \frac{\epsilon}{3}} \right) \tag{A.1}$$

Taking the union bound over the K communities and applying

$$|\mathcal{E}_k| \geq \pi_0 n$$

for all k, we have

$$\begin{aligned}
&\mathbb{P}\left(\frac{|R(e)-R^0(e)|\pi_0^\alpha}{n^{2-\alpha}} \geq K\epsilon\right) \\
=&\mathbb{P}\left(\left|\sum_{k=1}^K \frac{\pi_0^\alpha}{|\mathcal{E}_k|^\alpha n^{2-\alpha}} \sum_{i,j\in\mathcal{E}_k} \left(A_{ij}-\rho_n P_{c_i c_j}\right)\right| \geq K\epsilon\right) \\
\leq&\sum_{k=1}^K \mathbb{P}\left(\left|\frac{\pi_0^\alpha}{|\mathcal{E}_k|^\alpha n^{2-\alpha}} \sum_{i,j\in\mathcal{E}_k} \left(A_{ij}-\rho_n P_{c_i c_j}\right)\right| \geq \epsilon\right) \\
\leq&\sum_{k=1}^K \mathbb{P}\left(\left|\frac{1}{n^2} \sum_{i,j\in\mathcal{E}_k} \left(A_{ij}-\rho_n P_{c_i c_j}\right)\right| \geq \epsilon\right) \leq 2K\exp\left(-\frac{\frac{1}{2}t^2}{\rho_n n^2+\frac{1}{3}t}\right)
\end{aligned}$$

where the last inequality is due to (A.1). Setting $\epsilon=\rho_n\epsilon_0$, we have

$$\begin{aligned}
\mathbb{P}\left(\frac{|R(e;\alpha)-R^0(e;\alpha)|\pi_0^\alpha}{\rho_n n^{2-\alpha}} \geq K\epsilon_0\right) &\leq 2K\exp\left(-\frac{\frac{1}{2}\rho_n n^2\epsilon_0^2}{1+\frac{\epsilon_0}{3}}\right) \\
&= 2K\exp\left(-C_1\rho_n n^2\right)
\end{aligned}$$

where $C_1=\dfrac{\frac{1}{2}\epsilon_0^2}{1+\frac{\epsilon_0}{3}}$. Finally, taking a union bound over community assignments $e\in\mathcal{E}^{\pi_0}$, we have

$$\begin{aligned}
\mathbb{P}\left(\max_{e\in\mathcal{E}^{\pi_0}} \frac{|R(e;\alpha)-R^0(e;\alpha)|\pi_0^\alpha}{\rho_n n^{2-\alpha}} \geq K\epsilon_0\right) &\leq 2K^{n+1}\exp\left(-C_1\rho_n n^2\right) \\
&= 2K\exp\left(-C_1(\rho_n n-\log K)n\right) \to 0
\end{aligned}$$

since $\rho_n n\to\infty$. □

We next show that c, up to a permutation of community labels, is the unique maximizer of $R^0(e;\alpha)$ under mild conditions.Define $U\in\mathbb{R}^{K\times K}$ by $U_{kl}=\sum_{i=1}^n 1[e_i=k,c_i=l]/n$, and let D be a diagonal $K\times K$ matrix with $\pi_1,\ldots,\pi_K$ on the diagonal, where $\pi_k=\sum_{i=1}^n 1[c_i=k]/n$ is the fraction of nodes in community $\mathcal{C}_k$. Roughly speaking, U is the confusion matrix between e and c, and $U=DO$ for a permutation matrix O means the estimation is perfect. Define

$$g(U;\alpha)=\sum_{k=1}^K \frac{\sum_{l=1}^K\sum_{l'=1}^K U_{kl}U_{kl'}P_{ll'}}{\left(\sum_{a=1}^K U_{ka}\right)^\alpha}\ .$$

Each estimated community assignment e induces a unique $U=U(e)$. It is not difficult to verify that

$$g\left(U(e);\alpha\right)=\sum_{k=1}^K \frac{\sum_{i,j\in\mathcal{E}_k} P_{c_i c_j}}{|\mathcal{E}_k|^\alpha n^{2-\alpha}}=\frac{R^0(e;\alpha)}{\rho_n n^{2-\alpha}}\ .$$

and

$$2|e_1 - e_2| = \|U_1 - U_2\|_1$$

where $|e_1 - e_2|$ denotes the Hamming distance between e_1 and e_2.

Lemma 3. *Under Condition 2, if $\alpha \in [\max_{1\leq k<l\leq K} 2(K-1)P_{kl}/\min(P_{kk}, P_{ll}), 1]$, then for all U satisfying $\sum_{k=1}^K U_{kl} = \pi_l$ for $1 \leq k \leq K$, $g(U)$ is uniquely maximized at $U = DO$ for $O \in \mathcal{O}_K$, where $\mathcal{O}_K$ denotes the set of $K \times K$ permutation matrices.*

Proof of Lemma 3. We have

$$\begin{aligned}
&g(D;\alpha) - g(U;\alpha)\\
&\quad= \sum_{l=1}^K \left(\sum_{k=1}^K U_{kl}\right)^{2-\alpha} P_{ll} - \sum_{k=1}^K \frac{\sum_{l=1}^K U_{kl}^2 P_{ll} + \sum_{l=1}^K \sum_{l'\neq l} U_{kl}U_{kl'}P_{ll'}}{\left(\sum_{a=1}^K U_{ka}\right)^{\alpha}}\\
&\quad= \sum_{l=1}^K \left\{\left(\sum_{k=1}^K U_{kl}\right)^{2-\alpha} - \sum_{k=1}^K \frac{U_{kl}^2}{\left(\sum_{a=1}^K U_{ka}\right)^{\alpha}}\right\} P_{ll}\\
&\qquad- \sum_{k=1}^K \sum_{l=1}^K \sum_{l'\neq l} \left\{\frac{U_{kl}U_{kl'}}{\left(\sum_{a=1}^K U_{ka}\right)^{\alpha}}\right\} P_{ll'} \qquad (A.2)
\end{aligned}$$

For $0 < \alpha \leq 1$, since $U_{kl} \geq 0$ for all k and l, we have $\left(\sum_{k=1}^K U_{kl}\right)^{2-\alpha} \geq \sum_{k=1}^K U_{kl}^{2-\alpha}$. By mid-value theorem, there exists $\xi_{kl} \in \left(0, \sum_{a\neq l} U_{ka}\right)$, such that

$$\begin{aligned}
\left(\sum_{a=1}^K U_{ka}\right)^{\alpha} - U_{kl}^{\alpha} &= \alpha \left(\sum_{a\neq l} U_{ka}\right) / \left(U_{kl} + \xi^{kl}\right)^{1-\alpha}\\
&\geq \alpha \left(\sum_{a\neq l} U_{ka}\right) / \left(\sum_{a=1}^K U_{ka}\right)^{1-\alpha}. \qquad (A.3)
\end{aligned}$$

Finally, we will need the following inequality: for $0 < \alpha \leq 2$ and $x, y \geq 0$ satisfying $x + y \leq u$,

$$x^{2-\alpha}(u-x) + y^{2-\alpha}(u-y) \geq xyu^{1-\alpha}\ . \qquad (A.4)$$

For $x = y = 0$, equality holds. To verify (A.4) when $0 < x + y \leq u$, dividing by $u^{3-\alpha}$ we have

$$\begin{aligned}
&\frac{x^{2-\alpha}(u-x) + y^{2-\alpha}(u-y) - xyu^{1-\alpha}}{u^{3-\alpha}}\\
&= \left(\frac{x}{u}\right)^{2-\alpha}\left(1 - \frac{x}{u}\right) + \left(\frac{y}{u}\right)^{2-\alpha}\left(1 - \frac{y}{u}\right) - \frac{xy}{u^2}\\
&\geq \left(\frac{x}{u}\right)^{2}\left(1 - \frac{x}{u}\right) + \left(\frac{y}{u}\right)^{2}\left(1 - \frac{y}{u}\right) - \frac{xy}{u^2}
\end{aligned}$$

$$= \left\{ \left(\frac{x}{u}\right)^2 + \left(\frac{y}{u}\right)^2 - \frac{xy}{u^2} \right\} \left(1 - \frac{x+y}{u}\right) \geq 0 \ .$$

The first inequality above implies that a necessary condition for equality to hold in (A.4) is $xy = 0$.

We now lower bound the first term on the right hand side of (A.2).

$$\begin{aligned}
&\sum_{l=1}^{K} \left\{ \left(\sum_{k=1}^{K} U_{kl} \right)^{2-\alpha} - \sum_{k=1}^{K} \frac{U_{kl}^2}{\left(\sum_{a=1}^{K} U_{ka}\right)^{\alpha}} \right\} P_{ll} \\
&\geq \sum_{l=1}^{K} \sum_{k=1}^{K} \frac{U_{kl}^{2-\alpha} \left\{ \left(\sum_{a=1}^{K} U_{ka}\right)^{\alpha} - U_{kl}^{\alpha} \right\}}{\left(\sum_{a=1}^{K} U_{ka}\right)^{\alpha}} P_{ll} \\
&\geq \sum_{l=1}^{K} \sum_{k=1}^{K} \frac{U_{kl}^{2-\alpha} \left(\sum_{a \neq l} U_{ka}\right)}{\sum_{a=1}^{K} U_{ka}} \alpha P_{ll} \geq \sum_{l=1}^{K} \sum_{k=1}^{K} \frac{U_{kl}^{2-\alpha} \left(\sum_{a \neq l} U_{ka}\right)}{\sum_{a=1}^{K} U_{ka}} \sum_{l' \neq l} 2 P_{ll'} \\
&= \sum_{k=1}^{K} \left\{ \sum_{l=1}^{K} \sum_{l' \neq l} \frac{U_{kl}^{2-\alpha} \left(\sum_{a \neq l} U_{ka}\right) P_{ll'}}{\sum_{a=1}^{K} U_{ka}} + \sum_{l'=1}^{K} \sum_{l \neq l'} \frac{U_{kl'}^{2-\alpha} \left(\sum_{a \neq l'} U_{ka}\right) P_{ll'}}{\sum_{a=1}^{K} U_{ka}} \right\} \\
&= \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{l' \neq l} \frac{U_{kl}^{2-\alpha} \left(\sum_{a \neq l} U_{ka}\right) + U_{kl'}^{2-\alpha} \left(\sum_{a \neq l'} U_{ka}\right)}{\sum_{a=1}^{K} U_{ka}} P_{ll'} \\
&\geq \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{l' \neq l} \frac{U_{kl} U_{kl'}}{\left(\sum_{a=1}^{K} U_{ka}\right)^{\alpha}} P_{ll'} \ ,
\end{aligned} \tag{A.5}$$

where the last equality is obtained by applying (A.4) with $x = U_{kl}$, $y = U_{kl'}$ and $u = \sum_{a=1}^{K} U_{ka}$. Plugging (A.5) into (A.2), we have

$$g(D;\alpha) - g(U;\alpha) \geq 0 \ .$$

It remains to show that equality holds only if $U = DO$ for some $O \in \mathcal{O}_K$. Note that the last inequality in (A.5) is obtained from (A.4), where equality holds only when $xy = 0$. The corresponding condition for equality to hold in (A.5) is thus $U_{kl}U_{kl'} = 0$ for all k, l and l'. Therefore, for each k, there is only one l such that $U_{kl} \neq 0$, i.e., $U = DO$ for some $O \in \mathcal{O}_K$. □

Proof of Theorem 1. For any $\delta > 0$, define the set $\mathcal{U}_\delta$ as follows:

$$\mathcal{U}_\delta = \{U : \min_{O \in \mathcal{O}_K} \|U - OD\|_1 \geq \delta\} \ .$$

Since $\mathcal{U}_\delta$ is compact, by Lemma 3 we have

$$\max_{U \in \mathcal{U}_\delta} g(U;\alpha) < g(D;\alpha) \ .$$

The event

$$\Omega_\delta = \left\{ \max_{e \in \mathcal{E}^{\pi_0}} \frac{|R(e) - R^0(e)|}{\rho_n n^{2-\alpha}} < \frac{g(D;\alpha) - \max_{U \in \mathcal{U}_\delta} g(U;\alpha)}{2} \right\}$$

implies that the $\hat{e}$ that maximizes $g(U(e);\alpha)$ is not in $\mathcal{U}_\delta$, because

$$\begin{aligned} g(U(\hat{e});\alpha) = \frac{R^0(\hat{e})}{\rho_n n^{2-\alpha}} &> \frac{R(\hat{e})}{\rho_n n^{2-\alpha}} - \frac{g(D;\alpha) - \max_{U \in \mathcal{U}_\delta} g(U;\alpha)}{2} \\ \geq & \frac{R(c)}{\rho_n n^{2-\alpha}} - \frac{g(D;\alpha) - \max_{U \in \mathcal{U}_\delta} g(U;\alpha)}{2} \\ > & \frac{R^0(c)}{\rho_n n^{2-\alpha}} - \left(g(D;\alpha) - \max_{U \in \mathcal{U}_\delta} g(U;\alpha) \right) = \max_{U \in \mathcal{U}_\delta} g(U;\alpha) \end{aligned}$$

That is,

$$|\hat{e} - c| \leq \delta/2$$

By Lemma 2,

$$\mathbb{P}(\Omega_\delta) \to 1$$

We have shown that $\hat{e}$ converges to c in probability. □

Acknowledgments

This work was carried out while the first author was a PhD student at the University of Michigan. E.L. is partially supported by NSF grants DMS-1106772, DMS-1159005 and DMS-1521551. J.Z. is partially supported by a NSF grant DMS-1407698 and a NIH grant R01GM096194. The authors thank the Editor and three referees for their comments and suggestions that led to the improvement of this paper.

References

[1] Arash A Amini, Aiyou Chen, Peter J Bickel, and Elizaveta Levina. Pseudo-likelihood methods for community detection in large sparse networks. *The Annals of Statistics*, 41(4):2097–2122, 2013. MR3127859

[2] Norbert Binkiewicz, Joshua T Vogelstein, and Karl Rohe. Covariate assisted spectral clustering. *arXiv preprint arXiv:1411.2158*, 2014.

[3] H Cheng, Y Zhou, and J. X. Yu. Clustering large attributed graphs: A balance between structural and attribute similarities. *ACM Trans. Knowl. Discov. Data*, 5(2):12:1–12:33, February 2011.

[4] Wouter De Nooy, Andrej Mrvar, and Vladimir Batagelj. *Exploratory social network analysis with Pajek*, volume 27. Cambridge University Press, 2011.

[5] F. Glover. Future paths for integer programming and links to artificial intelligence. *Comput. Oper. Res.*, 13(5):533–549, May 1986. MR0868908

[6] M. S. Handcock, A. E. Raftery, and J. M. Tantrum. Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 170(2):301–354, 2007. MR2364300

[7] Tuan-Anh Hoang and Ee-Peng Lim. On joint modeling of topical communities and personal interest in microblogs. In *Social Informatics*, pages 1–16. Springer, 2014.

[8] P. D. Hoff. Random effects models for network data. In *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers*, pages 303–312. National Academies Press Washington, DC, 2003. MR2723712

[9] P. D. Hoff. Modeling homophily and stochastic equivalence in symmetric relational data. In *Advances in Neural Information Processing Systems*, volume 19. MIT Press, Cambridge, MA, 2007.

[10] P. W. Holland, K. B. Laskey, and S. Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109–137, 1983. MR0718088

[11] N. P. Hummon, P. Doreian, and L. C. Freeman. Analyzing the structure of the centrality-productivity literature created between 1948 and 1979. *Science Communication*, 11(4):459–480, 1990.

[12] B. Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. *Physical Review E*, 83:016107, 2011. MR2788206

[13] E. Lazega. *The collegial phenomenon: The social mechanisms of cooperation among peers in a corporate law partnership*. Oxford University Press, 2001.

[14] J. Leskovec M. Kim. Latent multi-group membership graph model. *International Conference on Machine Learning*, 2012.

[15] J. McAuley and J. Leskovec. Learning to discover social circles in ego networks. In *Advances in Neural Information Processing Systems 25*, pages 548–556, 2012.

[16] M. E. J. Newman. Modularity and community structure in networks. *Proc. Natl. Acad. Sci. USA*, 103(23):8577–8582, 2006.

[17] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69(2):026113, Feb 2004.

[18] MEJ Newman and Aaron Clauset. Structure and inference in annotated networks. *arXiv preprint arXiv:1507.04001*, 2015.

[19] Tai Qin and Karl Rohe. Regularized spectral clustering under the degree-corrected stochastic blockmodel. pages 3120–3128, 2013.

[20] E. M. Rogers and D. L. Kincaid. *Communication networks: Toward a new paradigm for research*. Free Press New York, 1981.

[21] T. Schlitt and A. Brazma. Current approaches to gene regulatory network modelling. *BMC Bioinformatics*, 8(Suppl 6):S9, 2007.

[22] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[23] A. Silva, W. Meira, Jr., and M. J. Zaki. Mining attribute-structure correlated patterns in large attributed graphs. *Proc. VLDB Endow.*, 5(5):466–477, 2012.

[24] Laura M Smith, Linhong Zhu, Kristina Lerman, and Allon G Percus. Partitioning networks with node attributes by compressing information flow. *arXiv preprint arXiv:1405.4332*, 2014.

[25] Tom AB Snijders, Philippa E Pattison, Garry L Robins, and Mark S Handcock. New specifications for exponential random graph models. *Sociological Methodology*, 36(1):99–153, 2006.

[26] C. Steglich, T. A. B. Snijders, and P. West. Applying siena: An illustrative analysis of the coevolution of adolescents' friendship networks, taste in music, and alcohol consumption. *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences*, 2(1):48, 2006.

[27] E. Viennet et al. Community detection based on structural and attribute similarities. In *ICDS 2012, The Sixth International Conference on Digital Society*, pages 7–12, 2012.

[28] Zhiqiang Xu, Yiping Ke, Yi Wang, Hong Cheng, and James Cheng. A model-based approach to attributed graph clustering. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, pages 505–516. ACM, 2012.

[29] Jaewon Yang, Julian McAuley, and Jure Leskovec. Community detection in networks with node attributes. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 1151–1156. IEEE, 2013.

[30] T. Yang, R. Jin, Y. Chi, and S. Zhu. Combining link and content for community detection: A discriminative approach. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 927–936. ACM, 2009.

[31] H. Zanghi, S. Volant, and C. Ambroise. Clustering based on random graph model embedding vertex features. *Pattern Recogn. Lett.*, 31(9):830–836, July 2010.