

Statistica Sinica Preprint No: SS-2017-0399

Title	Modularity Based Community Detection in Heterogeneous Networks
Manuscript ID	SS-2017-0399
URL	http://www.stat.sinica.edu.tw/statistica/
DOI	10.5705/ss.202017.0399
Complete List of Authors	Jingfei Zhang and Yuguo Chen
Corresponding Author	Jingfei Zhang
E-mail	ezhang@bus.miami.edu
Notice: Accepted version subject to English editing.	

MODULARITY BASED COMMUNITY DETECTION IN HETEROGENEOUS NETWORKS

Jingfei Zhang and Yuguo Chen

University of Miami and University of Illinois at Urbana-Champaign

Abstract: Heterogeneous networks are networks consisting of different types of nodes and multiple types of edges linking such nodes. While community detection has been extensively developed as a useful technique for analyzing networks that contain only one type of nodes, very few community detection techniques have been developed for heterogeneous networks. In this paper, we propose a modularity based community detection framework for heterogeneous networks. Unlike existing methods, the proposed approach has the flexibility to treat the number of communities as an unknown quantity. We describe a Louvain type maximization method for finding the community structure that maximizes the modularity function. Our simulation results show the advantages of the proposed method over existing methods. Moreover, the proposed modularity function is shown to be consistent under a heterogeneous stochastic blockmodel framework. Analyses of the DBLP four-area dataset and a MovieLens dataset demonstrate the usefulness of the proposed method.

Key words and phrases: Heterogeneous network, modularity function, community detection, null model, consistency.

1. Introduction

Network community detection has attracted much recent attention from various scientific communities, including statistics, physics, information technology, biology, social science and many others. A real-world network often displays a high level of inhomogeneity in its edge distribution, with high edge density within certain groups of nodes and low edge density be-

tween these groups. This feature is often referred to as the *community structure* (Fortunato, 2010). Community structures have been observed in networks in social science, biology, political science and so on. For example, in a gene regulation network, communities are groups of genes that function together in biological processes to carry out specific functions (Zhang and Cao, 2018). Detecting communities in real-world networks can help us better understand the architecture of the network. Further, it allows us to investigate the property in individual communities, which may be different from the aggregated property from the network as a whole.

Many community detection techniques have been proposed in recent years. See Fortunato (2010) for a comprehensive review. One class of methods involve maximizing some partition quality function over all possible partitions of the network (Shi and Malik, 2000; Newman and Girvan, 2004; Newman, 2006). One class of methods use spectral clustering techniques (Rohe et al., 2011; Rohe et al., 2016). Another class includes model based approaches that estimate community structures through fitting probabilistic models to the observed networks (Airoldi et al., 2008; Bickel and Chen, 2009; Jin, 2015). In the second and third classes of approaches, we need to know the number of communities *a priori*.

The existing community detection approaches primarily focus on homogeneous networks, i.e., networks with only one type of nodes. However, networks representing real-world complex systems often contain different types of nodes and different types of edges linking such nodes; we refer to such type of networks as *heterogeneous networks*. For example, in a health-care network, nodes can be patients, diseases, doctors and hospitals. The edges can be in the type of patient-disease (patient treated for disease), patient-doctor (patient treated by doctor), doctor-hospital (doctor works at hospital). Figure 1.1 provides a simple illustration

of a heterogeneous network. In this illustrative heterogeneous Facebook network, there are two types of nodes, users and events. Furthermore, there are two types of interactions in this network. A user is linked to another user through friendship and a user is linked to an event through attendance.

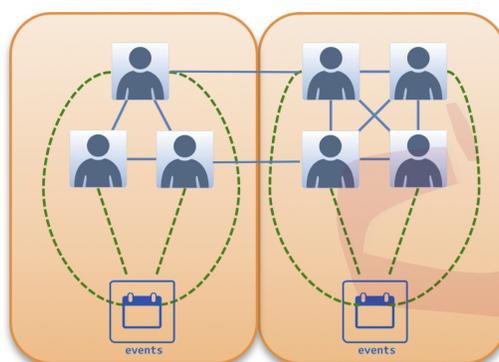


Figure 1.1: Two communities in a heterogeneous Facebook network with two types of nodes: users and events.

To find communities in a heterogeneous network using the existing methods developed for homogeneous networks, there are two possible approaches. One approach is to treat the heterogeneous network as a homogeneous network. In this approach, we do not differentiate the different types of nodes and edges. The other approach is to consider each type of nodes in the network separately, i.e., discard the information from the edges linking different types of nodes. In both approaches, we lose important information. In the first approach, we ignore the fact that different types of nodes may behave differently. For example, in Figure 1.1, users and events behave in different ways; a user can become friends with other users but an event can not link to other events. Using the first approach, the community detection algorithm does not distinguish the two different types of nodes. Losing such important information may lead to poor community detection results. In the second approach, the

valuable information from edges that link different types of nodes are ignored. For example, in Figure 1.1, the user-event links show how different users are attracted to different events. Including such information can help us better identify the communities in users. Moreover, it provides important insights on the types of events that are clustered with each community of users.

To find community structures in heterogeneous networks, a preferable approach should take into account all the information contained in the heterogeneous network, including the different types of nodes, the homogeneous edges (edges that connect two nodes of the same type) and the heterogeneous edges (edges that connect two nodes of different types). The objective of the approach is to cluster the nodes in the heterogeneous network into several non-overlapping groups such that there are more homogeneous and heterogeneous edges within these groups and fewer homogeneous and heterogeneous edges between these groups; see Figure 1.1 for a simple illustration of a heterogeneous Facebook network with two communities.

Several methods have been proposed recently for detecting communities in heterogeneous networks (Sun and Han, 2012; Liu et al., 2014; Sengupta and Chen, 2015). One limitation of the existing methods is that they may have requirements on the number of node types or edge types in the network (see for example Sun and Han (2012) and Liu et al. (2014)). Another limitation of the existing methods is that they may require the number of communities in the network to be pre-specified (see for example Sengupta and Chen (2015)). This requirement could be difficult to meet in practice, since we generally do not know the number of communities in real-world networks. Lastly, very large networks can be computationally challenging for some existing methods, such as the spectral clustering approach proposed in

Sengupta and Chen (2015).

In this paper, we propose a modularity based heterogeneous network community detection framework. Our contribution is threefold. First, we formally define a null model for a heterogeneous network. Under the proposed null model, we calculate the probabilities of having a homogeneous edge between two nodes of the same type and a heterogeneous edge between two nodes of different types. Second, we propose a Louvain type maximization method that can efficiently maximize the proposed modularity function. The application of the maximization method on a real-world network with about 20,000 nodes takes less than 20 seconds on a standard PC. Our proposed approach can be applied to heterogeneous networks of any type. Furthermore, the number of communities does not need to be specified and can be treated as an unknown quantity. Third, we show that the proposed modularity function for heterogeneous networks is consistent under a heterogeneous stochastic blockmodel framework. The consistency properties of modularity functions formulated for bipartite or multipartite networks follow as special cases. This theoretical result fills an existing gap in the literature.

The rest of this article is organized as follows. Section 2 introduces the null model for a heterogeneous network and the definition of a modularity function. Section 3 discusses the Louvain type modularity maximization technique. Section 4 shows the consistency of the modularity function under the a heterogeneous stochastic blockmodel framework. Section 5 demonstrates the advantages of our proposed method through simulation studies. Section 6 discusses the application of the proposed method on the DBLP four-area dataset and a MovieLens dataset. Section 7 provides some concluding remarks and discussions.

2. Modularity Function for Heterogeneous Networks

Let $\mathcal{G} = (\bigcup_{i=1}^L V^{[i]}, \mathcal{E} \cup \mathcal{E}^+)$ denote a simple heterogeneous network (no self loops or multiple edges) with L types of nodes. Node set $V^{[i]}$ contains all the nodes of the i th type, $i = 1, \dots, L$. Edge set \mathcal{E} denotes the set of edges between nodes of the same type and \mathcal{E}^+ denotes the set of edges between nodes of different types. A homogeneous network $G_i(V^{[i]}, E^{[i]})$ can be formed within each node set $V^{[i]}$, where $E^{[i]}$ is the set of edges between nodes in $V^{[i]}$. By definition, we have $\mathcal{E} = \bigcup_{i=1}^L E^{[i]}$. When $\mathcal{E} = \emptyset$, the heterogeneous network \mathcal{G} forms a multi-partite network, i.e., edges are only established between different types of nodes. In this paper, we use the terms *network* and *graph* interchangeably.

Newman and Girvan (2004) defined a quality function, usually referred to as the *modularity function*, for measuring the strength of a division of a homogeneous network into communities. Given a homogeneous network $G(V, E)$ with n nodes, m edges and a community assignment $\mathbf{e} = (e_1, \dots, e_n)$, where $e_i \in \{1, \dots, K\}$ is the community that node i belongs to, the modularity function is defined as

$$Q(\mathbf{e}, G) = \frac{1}{2m} \sum_{i,j} [A_{ij} - E(A_{ij})] \delta(e_i, e_j), \quad (2.1)$$

where $\delta(r, s) = 1$ if $r = s$ and 0 otherwise. Here A_{ij} is the (i, j) th entry of the adjacency matrix A of the network and the expectation $E(A_{ij})$ is calculated under some null model that describes networks with no community structure. It is easy to see that $Q(\mathbf{e}, G) \in [-1, 1]$.

The modularity function for homogeneous networks measures the difference between the observed number of intra-community edges and the expected number of intra-community edges under the null model. If the observed number of intra-community edges in the network is close to the expected value, the modularity Q is close to 0. When Q approaches 1, the observed number of intra-community edges is much higher than the expected value, and this indicates a strong community structure. Since the modularity function measures the

“strength” of community structure with respect to a network partition, the community membership of a network is identified by maximizing the modularity function $Q(\mathbf{e}, G)$ with respect to \mathbf{e} . The number of communities K does not need to be pre-specified in this approach and can be treated as an unknown quantity.

To introduce the modularity based community detection framework for heterogeneous networks, we focus on the case with only two types of nodes ($L = 2$). The framework can be easily generalized to networks that contain more than two types of nodes. For a heterogeneous network $\mathcal{G} = (V^{[1]} \cup V^{[2]}, \mathcal{E} \cup \mathcal{E}^+)$, let $G_1 = (V^{[1]}, E^{[1]})$ and $G_2 = (V^{[2]}, E^{[2]})$ denote the two homogeneous networks within node sets $V^{[1]} = (v_1^{[1]}, \dots, v_{n_1}^{[1]})$ and $V^{[2]} = (v_1^{[2]}, \dots, v_{n_2}^{[2]})$, respectively. Furthermore, let $G_{12} = (V^{[1]} \cup V^{[2]}, \mathcal{E}^+)$ denote the bi-partite network formed between node sets $V^{[1]}$ and $V^{[2]}$. We subsequently refer to nodes in $V^{[1]}$ ($V^{[2]}$) as type-[1] (type-[2]) nodes, edges in $E^{[1]}$ ($E^{[2]}$) as type-[1] (type-[2]) edges, and edges in $E^{[12]}$ as type-[12] edges. We consider the following three matrices:

- $A^{[1]}$, the $n_1 \times n_1$ 0-1 adjacency matrix of $G_1 = (V^{[1]}, E^{[1]})$, where $A_{ij}^{[1]} = 1$ if and only if there is an edge between $v_i^{[1]}$ and $v_j^{[1]}$.
- $A^{[2]}$, the $n_2 \times n_2$ 0-1 adjacency matrix of $G_2 = (V^{[2]}, E^{[2]})$, where $A_{ij}^{[2]} = 1$ if and only if there is an edge between $v_i^{[2]}$ and $v_j^{[2]}$.
- $A^{[12]}$, the $n_1 \times n_2$ 0-1 matrix of $G_{12} = (V^{[1]} \cup V^{[2]}, \mathcal{E}^+)$, where $A_{ij}^{[12]} = 1$ if and only if there is an edge between $v_i^{[1]}$ and $v_j^{[2]}$.

Note that $A^{[12]}$ is not the adjacency matrix of $G_{12} = (V^{[1]} \cup V^{[2]}, \mathcal{E}^+)$, but only a submatrix of it. The adjacency matrix of G_{12} is

$$\begin{pmatrix} \mathbf{0} & A^{[12]} \\ A^{[21]} & \mathbf{0} \end{pmatrix},$$

where $A^{[21]} = A^{[12]T}$. We use A^T to denote the transpose of matrix A . The matrix $A^{[12]}$ is usually referred to as the *bi-adjacency matrix* of G_{12} . Since we only focus on networks with undirected edges, adjacency matrices $A^{[1]}$ and $A^{[2]}$ are both symmetric. The heterogeneous network \mathcal{G} can be uniquely represented by its $(n_1 + n_2) \times (n_1 + n_2)$ adjacency matrix \mathcal{A} ,

$$\mathcal{A} = \begin{pmatrix} A^{[1]} & A^{[12]} \\ A^{[21]} & A^{[2]} \end{pmatrix}.$$

2.1. Null Model for Heterogeneous Networks

The modularity function measures the difference between the observed network and the null model that characterizes networks with no community structure. To define the modularity function for a heterogeneous network, we need to formulate a null model for heterogeneous networks.

We introduce the following notations on degree sequences:

- $\mathbf{d}^{[1]} = (d_1^{[1]}, \dots, d_{n_1}^{[1]})$, where $d_i^{[1]} = \sum_{j=1}^{n_1} A_{ij}^{[1]}$, $i = 1, \dots, n_1$, is the number of links incident to $v_i^{[1]}$ from $V^{[1]}$.
- $\mathbf{d}^{[2]} = (d_1^{[2]}, \dots, d_{n_2}^{[2]})$, where $d_i^{[2]} = \sum_{j=1}^{n_2} A_{ij}^{[2]}$, $i = 1, \dots, n_2$, is the number of links incident to $v_i^{[2]}$ from $V^{[2]}$.
- $\mathbf{d}^{[12]} = (d_1^{[12]}, \dots, d_{n_1}^{[12]})$, where $d_i^{[12]} = \sum_{j=1}^{n_2} A_{ij}^{[12]}$, $i = 1, \dots, n_1$, is the number of links incident to $v_i^{[1]}$ from $V^{[2]}$.
- $\mathbf{d}^{[21]} = (d_1^{[21]}, \dots, d_{n_2}^{[21]})$, where $d_i^{[21]} = \sum_{j=1}^{n_1} A_{ij}^{[21]}$, $i = 1, \dots, n_2$, is the number of links incident to $v_i^{[2]}$ from $V^{[1]}$.

From the definitions, we see that $\mathbf{d}^{[1]}$ is the vector of column (row) sums of $A^{[1]}$, $\mathbf{d}^{[12]}$ is the vector of row sums of $A^{[12]}$, $\mathbf{d}^{[21]}$ is the vector of column sums of $A^{[12]}$, and $\mathbf{d}^{[2]}$ is the vector of column (row) sums of $A^{[2]}$. Write the number of edges in G_1 as $m^{[1]} = \sum_{i=1}^{n_1} d_i^{[1]}/2$, the number

of edges in G_{12} as $m^{[12]} = \sum_{i=1}^{n_1} d_i^{[12]}$, and the number of edges in G_2 as $m^{[2]} = \sum_{i=1}^{n_2} d_i^{[2]}/2$. Define $\mathcal{D} = (\mathbf{d}^{[1]}, \mathbf{d}^{[12]}, \mathbf{d}^{[2]}, \mathbf{d}^{[21]})$.

An appropriate null model should satisfy the following two conditions. First, it should describe a random heterogeneous network with no community structure. Second, the networks from the null model should share basic structural properties with the observed network (Newman, 2006; Zhang and Chen, 2016). For the null model of a heterogeneous network, we propose to preserve the observed degree sequence $(\mathbf{d}^{[1]}, \mathbf{d}^{[12]}, \mathbf{d}^{[2]}, \mathbf{d}^{[21]})$. That is, the degrees $d_i^{[1]}$ and $d_i^{[12]}$ for each node $v_i^{[1]}$, $i = 1 \dots, n_1$, are fixed. Similarly, the degrees $d_i^{[2]}$ and $d_i^{[21]}$ for each node $v_i^{[2]}$, $i = 1 \dots, n_2$, are fixed.

Preserving the observed degree sequence has been considered in various homogeneous network models in the literature (Chung and Lu, 2002; Newman and Girvan, 2004; Perry and Wolfe, 2012). The edge distribution in real-world networks often displays high global inhomogeneity and local inhomogeneity. The global inhomogeneity refers to the feature that most nodes have low degrees while a few nodes have high degrees. The local inhomogeneity refers to the case that there is a high concentration of edges within certain groups of nodes and low concentration of edges between these groups. The local inhomogeneity is also referred to as the community structure. To study the local inhomogeneity, it is important to control for the global inhomogeneity. That is, to study the community structure, it is important to control for the degree sequence.

The sample space in our null model is defined as

$$\Sigma_{\mathcal{D}} = \{\mathcal{G} : \mathcal{G} \text{ is a simple heterogeneous network with degree sequence } \mathcal{D}\}.$$

For a heterogeneous network \mathcal{G} from the sample space, the null distribution is defined as

$$p(\mathcal{G}) = \frac{1}{|\Sigma_{\mathcal{D}}|}. \quad (2.2)$$

Under the null model, every heterogeneous network from $\Sigma_{\mathcal{D}}$ is equally likely to occur and there is no preference for any network configuration. With the defined null model, we need to calculate the expectations $E_p(A_{ij}^{[1]})$, $E_p(A_{ij}^{[2]})$ and $E_p(A_{ij}^{[2]})$ for the modularity function defined in the Section 2.2. Here the expectation $E_p(\cdot)$ is taken with respect to $p(\cdot)$ in (2.2).

To calculate $E(A_{ij}^{[l]})$ under the null model, we notice that

$$E(A_{ij}^{[l]}) = \frac{|\Sigma_{\mathcal{D}|A_{ij}^{[l]}=1}|}{|\Sigma_{\mathcal{D}}|},$$

where $\Sigma_{\mathcal{D}|A_{ij}^{[l]}=1}$ is the set of all simple heterogeneous networks in $\Sigma_{\mathcal{D}}$ with $A_{ij}^{[l]}=1$, $l = 1, 2$. Denote $\Sigma_{\mathbf{d}^{[1]}}$ as the set of all simple homogeneous graphs with degree sequence $\mathbf{d}^{[1]}$, $\Sigma_{\mathbf{d}^{[2]}}$ as the set of all simple homogeneous graphs with degree sequence $\mathbf{d}^{[2]}$, and $\Sigma_{\mathbf{d}^{[12]}, \mathbf{d}^{[21]}}$ as the set of all bipartite graphs with degree sequence $\mathbf{d}^{[12]}$ for type-[1] nodes and degree sequence $\mathbf{d}^{[21]}$ for type-[2] nodes. We have $|\Sigma_{\mathcal{D}}| = |\Sigma_{\mathbf{d}^{[1]}}| \times |\Sigma_{\mathbf{d}^{[2]}}| \times |\Sigma_{\mathbf{d}^{[12]}, \mathbf{d}^{[21]}}|$. It is easy to see that

$$E(A_{ij}^{[l]}) = \frac{|\Sigma_{\mathbf{d}^{[l]}|A_{ij}^{[l]}=1}|}{|\Sigma_{\mathbf{d}^{[l]}}|}, \quad l = 1, 2, \quad (2.3)$$

where $|\Sigma_{\mathbf{d}^{[l]}|A_{ij}^{[l]}=1}|$ is the total number of simple homogeneous networks with degree sequence $\mathbf{d}^{[l]}$ and a link between nodes i and j . Similarly, we can show that

$$E(A_{ij}^{[12]}) = \frac{|\Sigma_{\mathbf{d}^{[12]}, \mathbf{d}^{[21]}|A_{ij}^{[12]}=1}|}{|\Sigma_{\mathbf{d}^{[12]}, \mathbf{d}^{[21]}}|}, \quad (2.4)$$

and

$$E(A_{ij}^{[21]}) = \frac{|\Sigma_{\mathbf{d}^{[12]}, \mathbf{d}^{[21]}|A_{ij}^{[21]}=1}|}{|\Sigma_{\mathbf{d}^{[12]}, \mathbf{d}^{[21]}}|}, \quad (2.5)$$

where $|\Sigma_{\mathbf{d}^{[12]}, \mathbf{d}^{[21]}|A_{ij}^{[12]}=1}|$ is the total number of bi-partite graphs with degree sequences $\mathbf{d}^{[12]}$ for type-[1] nodes, $\mathbf{d}^{[21]}$ for type-[2] nodes and a link between the i th node of type-[1] and the j th node of type-[2].

Calculating the numerators and the denominators in (2.3), (2.4) and (2.5) is a difficult problem. Bender and Canfield (1978) and McKay (2010) derived asymptotic formulae for

the number of simple graphs with a fixed degree sequence and pre-specified structure zeroes (a structure zero at A_{ij} means no edge can be placed between node i and node j). Based on these asymptotic formulae, we have the following approximations for the expectations.

Theorem 1. Define $d_{max}^{[l]} = \max_{i=1}^{n_l} d_i^{[l]}$, $l = 1, 2$, $d_{max}^{[12]} = \max_{i=1}^{n_1} d_i^{[12]}$, $d_{max}^{[21]} = \max_{i=1}^{n_2} d_i^{[21]}$.

Suppose that $d_{max}^{[l]} = o(m^{[l]1/2})$, $l = 1, 2$, $d_{max}^{[12]} = o(m^{[12]1/2})$, $d_{max}^{[21]} = o(m^{[21]1/2})$. We have

$$E(A_{ij}^{[l]}) = \frac{d_i^{[l]} d_j^{[l]}}{2m^{[l]}} (1 + o(1)), \quad E(A_{ij}^{[12]}) = \frac{d_i^{[12]} d_j^{[21]}}{m^{[12]}} (1 + o(1)), \quad l = 1, 2.$$

We refer to the online supplementary material for the proof. The conditions in Theorem 1 describe the density of the network as the network size tends to infinity. Since $d_{max}^{[l]} \geq 2m^{[l]}/n_l$, the condition $d_{max}^{[l]} = o(m^{[l]1/2})$ also implies that $d_{max}^{[l]} = o(n_l)$, which describes the rates that the maximum node degrees increase at; these conditions are to make sure the network does not become extremely dense as the network size grows. Similarly, we can derive that $d_{max}^{[12]} = o(n_1)$ and $d_{max}^{[21]} = o(n_2)$.

The results in Theorem 1 indicate that $E(A_{ij}^{[l]})$ can be well approximated by $\frac{d_i^{[l]} d_j^{[l]}}{2m^{[l]}}$ and $E(A_{ij}^{[12]})$ can be well approximated by $\frac{d_i^{[12]} d_j^{[21]}}{m^{[12]}}$. As such, we use these approximations in the modularity function defined in the next section.

2.2. Modularity Function

We first consider heterogeneous networks with only two types of nodes ($L = 2$). Later in this section, we generalize the results to heterogeneous networks with any $L \geq 2$. We define the $(n_1 + n_2) \times (n_1 + n_2)$ modularity matrix \mathcal{M} for the heterogeneous network \mathcal{G} as

$$\mathcal{M} = \begin{pmatrix} M^{[1]}/2m^{[1]} & M^{[12]}/m^{[12]} \\ M^{[21]}/m^{[21]} & M^{[2]}/2m^{[2]} \end{pmatrix},$$

where $M^{[1]} = A^{[1]} - E(A^{[1]})$, $M^{[2]} = A^{[2]} - E(A^{[2]})$, $M^{[12]} = A^{[12]} - E(A^{[12]})$ and $M^{[21]} = A^{[21]} - E(A^{[21]})$. If there are no edges within the type-[1] (or type-[2]) nodes, we set $M^{[1]}/2m^{[1]} =$

$\mathbf{0}_{n_1 \times n_1}$ (or $M^{[2]}/2m^{[2]} = \mathbf{0}_{n_2 \times n_2}$). Similarly, if there are no edges between type-[1] and type-[2] nodes, we set $M^{[12]}/m^{[12]} = \mathbf{0}_{n_1 \times n_2}$ and $M^{[21]}/m^{[21]} = \mathbf{0}_{n_2 \times n_1}$.

Define a 0-1 assignment matrix \mathcal{B} of dimension $(n_1 + n_2) \times K$ as

$$\mathcal{B} = \begin{pmatrix} B^{[1]} \\ B^{[2]} \end{pmatrix}, \quad (2.6)$$

where $B^{[1]}$ is an $n_1 \times K$ matrix with $B_{ij}^{[1]} = 1$ if node $v_i^{[1]}$ is in the j -th community and 0 otherwise, and $B^{[2]}$ is an $n_2 \times K$ matrix with $B_{ij}^{[2]} = 1$ if node $v_i^{[2]}$ is in the j -th community and 0 otherwise. The modularity function of a heterogeneous network is defined as

$$\begin{aligned} Q(\mathcal{B}, \mathcal{G}) &= \frac{1}{4} \text{tr}(\mathcal{B}^T \mathcal{M} \mathcal{B}) \\ &= \frac{1}{4} \left[\frac{1}{2m^{[1]}} \text{tr}(B^{[1]T} M^{[1]} B^{[1]}) + \frac{2}{m^{[12]}} \times \text{tr}(B^{[1]T} M^{[12]} B^{[2]}) + \frac{1}{2m^{[2]}} \text{tr}(B^{[2]T} M^{[2]} B^{[2]}) \right], \end{aligned} \quad (2.7)$$

where $\text{tr}(\cdot)$ denotes the trace of a square matrix. With some calculations, we can derive that

$$\begin{aligned} \frac{1}{2m^{[1]}} \text{tr}(B^{[1]T} M^{[1]} B^{[1]}) &= \frac{1}{2m^{[1]}} \sum_{i,j} [A_{ij}^{[1]} - E(A_{ij}^{[1]})] I(B_{i\cdot}^{[1]} = B_{j\cdot}^{[1]}), \\ \frac{2}{m^{[12]}} \text{tr}(B^{[1]T} M^{[12]} B^{[2]}) &= \frac{2}{m^{[12]}} \sum_{i,j} [A_{ij}^{[12]} - E(A_{ij}^{[12]})] I(B_{i\cdot}^{[1]} = B_{j\cdot}^{[2]}), \\ \frac{1}{2m^{[2]}} \text{tr}(B^{[2]T} M^{[2]} B^{[2]}) &= \frac{1}{2m^{[2]}} \sum_{i,j} [A_{ij}^{[2]} - E(A_{ij}^{[2]})] I(B_{i\cdot}^{[2]} = B_{j\cdot}^{[2]}). \end{aligned}$$

Here $B_{i\cdot}$ denotes the i th row of matrix B and $I(\cdot)$ is an indicator function. For example, $I(B_{i\cdot}^{[1]} = B_{j\cdot}^{[1]}) = 1$ only when nodes i and j are both of type-[1] and they are in the same community. The first component $\text{tr}(B^{[1]T} M^{[1]} B^{[1]})/2m^{[1]}$ and the third component $\text{tr}(B^{[2]T} M^{[2]} B^{[2]})/2m^{[2]}$ calculate the differences between the observed number of intra-community edges and the expected number of intra-community edges in networks G_1 and G_2 , respectively. The second component $\text{tr}(B^{[1]T} M^{[12]} B^{[2]})/m^{[12]}$ calculates the difference

between the observed number of intra-community edges and the expected number of intra-community edges in the bi-partite network G_{12} .

From the definition, we can see the modularity function $Q(\mathcal{B}, \mathcal{G}) \in [-1, 1]$. When $Q(\mathcal{B}, \mathcal{G})$ approaches 1, the observed numbers of type-[1], type-[2] and type-[12] intra-community edges are much higher than the expected values, which indicates a strong community structure. On the other hand, when $Q(\mathcal{B}, \mathcal{G})$ approaches 0, the observed numbers of type-[1], type-[2] and type-[12] intra-community edges are close to the expected values, which indicates a weak community structure. Note that when there is only one type of nodes in the network, the proposed modularity reduces to the Newman-Girvan modularity.

To generalize the modularity function to a heterogeneous network with L types of nodes, we denote the adjacency matrix of $G_i(V^{[i]}, E^{[i]})$ as $A^{[i]}$ and the bi-adjacency matrix of $G_{ij}(V^{[i]} \cup V^{[j]}, E^{[ij]})$ as $A^{[ij]}$, $1 \leq i \neq j \leq L$. Write the number of nodes in each type as $n_i = |V^{[i]}|$, $i = 1, \dots, L$. Further, write the number of edges in $G_i(V^{[i]}, E^{[i]})$ as $m^{[i]}$ and the number of edges in $G_{ij}(V^{[i]} \cup V^{[j]}, E^{[ij]})$ as $m^{[ij]}$, $1 \leq i \neq j \leq L$. The modularity function is defined as

$$Q(\mathcal{B}, \mathcal{G}) = \frac{1}{L^2} \text{tr}(\mathcal{B}^T \mathcal{M} \mathcal{B}), \quad (2.8)$$

where

$$\mathcal{M} = \begin{pmatrix} M^{[1]}/2m^{[1]} & \dots & M^{[1L]}/m^{[1L]} \\ \vdots & \ddots & \vdots \\ M^{[L1]}/m^{[L1]} & \dots & M^{[L]}/2m^{[L]} \end{pmatrix}, \quad \mathcal{B} = \begin{pmatrix} B^{[1]} \\ \vdots \\ B^{[L]} \end{pmatrix}.$$

Here $M^{[i]} = A^{[i]} - E(A^{[i]})$, $M^{[ij]} = A^{[ij]} - E(A^{[ij]})$, $1 \leq i \neq j \leq L$. Matrix \mathcal{B} is a $(n_1 + \dots + n_L) \times K$ assignment matrix defined similarly as that in (2.6). The expectations in the modularity function are approximated using the following corollary.

COROLLARY 1. Define $d_{max}^{[l_1]} = \max_{i=1}^{n_{l_1}} d_i^{[l_1]}$, $d_{max}^{[l_1 l_2]} = \max_{i=1}^{n_{l_1}} d_i^{[l_1 l_2]}$, $1 \leq l_1 \neq l_2 \leq L$. Suppose that $d_{max}^{[l_1]} = o(m^{[l_1]^{1/2}})$, $d_{max}^{[l_1 l_2]} = o(m^{[l_1 l_2]^{1/2}})$, $1 \leq l_1 \neq l_2 \leq L$. We have

$$E(A_{ij}^{[l_1]}) = \frac{d_i^{[l_1]} d_j^{[l_1]}}{2m^{[l_1]}}(1 + o(1)), \quad E(A_{ij}^{[l_1 l_2]}) = \frac{d_i^{[l_1 l_2]} d_j^{[l_1 l_2]}}{m^{[l_1 l_2]}}(1 + o(1)), \quad 1 \leq l_1 \neq l_2 \leq L.$$

The corollary is directly available from Theorem 1. Since a larger modularity value indicates a stronger community structure, the community assignment of nodes in the heterogeneous network \mathcal{G} is identified by maximizing the modularity function with respect to \mathcal{B} . In the next section, we introduce a Louvain type method for efficient modularity function maximization.

3. Modularity Maximization

Our goal is to find the community assignment matrix \mathcal{B} that maximizes the modularity function in (2.8), i.e.,

$$\arg \max_{\substack{\mathcal{B}_{(n_1 + \dots + n_L) \times K} \\ K \in \mathbb{Z}^+}} tr(\mathcal{B}^T \mathcal{M} \mathcal{B}).$$

Maximizing this objective function is a very difficult problem, especially since the number of communities K is generally unknown. Brandes et al. (2008) showed that finding the partition that maximizes the modularity function for a homogeneous network is NP-hard. Existing heuristic approaches for maximizing the modularity function come from various fields, including computer science, physics and sociology (Clauset et al., 2004; Massen and Doye, 2005; Newman, 2006; Reichardt and Bornholdt, 2006; Agrawal and Kempe, 2008). In this paper, we adopt a Louvain type maximization method.

The Louvain maximization method is a hierarchical clustering method proposed in Blondel et al. (2008). The technique was developed to maximize the modularity function of a homogeneous network. The optimization procedure is carried out in two phases that are

repeated iteratively. The first phase starts by assigning each node in the network to its own community (each community contains one and only one node). Then each node i is moved to the neighboring community that results in the largest increase in modularity (if no increase is possible, then node i remains in its original community). A neighboring community of node i is defined as a community that node i is linked to. In the second phase, the algorithm aggregates nodes in the same community and “constructs” a new network whose nodes are the communities from the first phase. The edges between the new nodes are calculated using the edges connecting the two corresponding communities (see Blondel et al. (2008) for details). These steps are repeated iteratively until the modularity reaches its local maximum.

The Louvain method were successfully applied to various homogeneous networks of sizes up to 100 million nodes and billions of links. Using the Louvain method for community detection in a typical network with 2 million nodes only takes several minutes on a standard PC (Blondel, 2011). Fortunato (2012) noted that the modularity maximum found by the Louvain method often compares favorably with those found by using the methods in Clauset et al. (2004) and Wakita and Tsurumi (2007).

Similar to the Louvain method, finding the maximizer of the proposed heterogeneous network modularity function can also be carried out in two phases that are repeated iteratively. To ease the presentation, we focus on the case where $L = 2$, i.e., there are two types of nodes. First we define a term “unit”. A unit may contain one node of any type or two nodes of different types. A community consists of several units. To initialize, we assign each node in the network to its own unit. Therefore, if there are n_1 type-[1] nodes and n_2 type-[2] nodes, the algorithm starts with $n_1 + n_2$ units. In the first phase, we start by assigning each unit to its own community. Then we calculate the change in modularity when unit i is assigned to

each one of its neighboring communities. A neighboring community of unit i is defined as a community that unit i is linked to. Once this value is calculated for every community that unit i is linked to, we assign unit i to the community that leads to the largest increase in modularity. If no move increases the modularity, unit i remains in its original community. This step is applied repeatedly to the units in the network until no increase in modularity can be achieved. In the second phase, we examine each community from the first phase and merge nodes of the same type in each community. This community then becomes a new unit in the next step. If two communities are linked, then there is an edge between the two new units; if two communities are not linked, then there is no edge between the two new units. We repeat these two phases iteratively until there is no move possible and the modularity reaches a local maximum.

As an example, Figure 3.2 shows the application of the proposed algorithm to a heterogeneous network with 2 types of nodes. Each iteration contains two phases. In the first iteration, the number of communities changes from 11 to 4. After the first iteration, nodes 1 and 2 are merged and treated as one node, say $v_{1,2}^*$, in the second iteration; similarly, nodes 7 and 8 are merged and treated as one node, say $v_{7,8}^*$; node 3 does not merge with any node and is treated as one node, say v_3^* . In the second iteration, nodes $\{v_3^*, v_{7,8}^*\}$ form a unit and node $v_{1,2}^*$ is a unit. During the first phase in the second iteration, we compute the change in modularity when we place unit $v_{1,2}^*$ and unit $\{v_3^*, v_{7,8}^*\}$ in one community. If the modularity increases, we place $v_{1,2}^*$ and $\{v_3^*, v_{7,8}^*\}$ in one community; if the modularity decreases, the two units remain in their original communities. In the second iteration, the number of communities changes from 4 to 2. The algorithm outputs two communities with the first community including nodes 1, 2, 3, 7, 8 and the second community including nodes

4, 5, 6, 9, 10, 11.

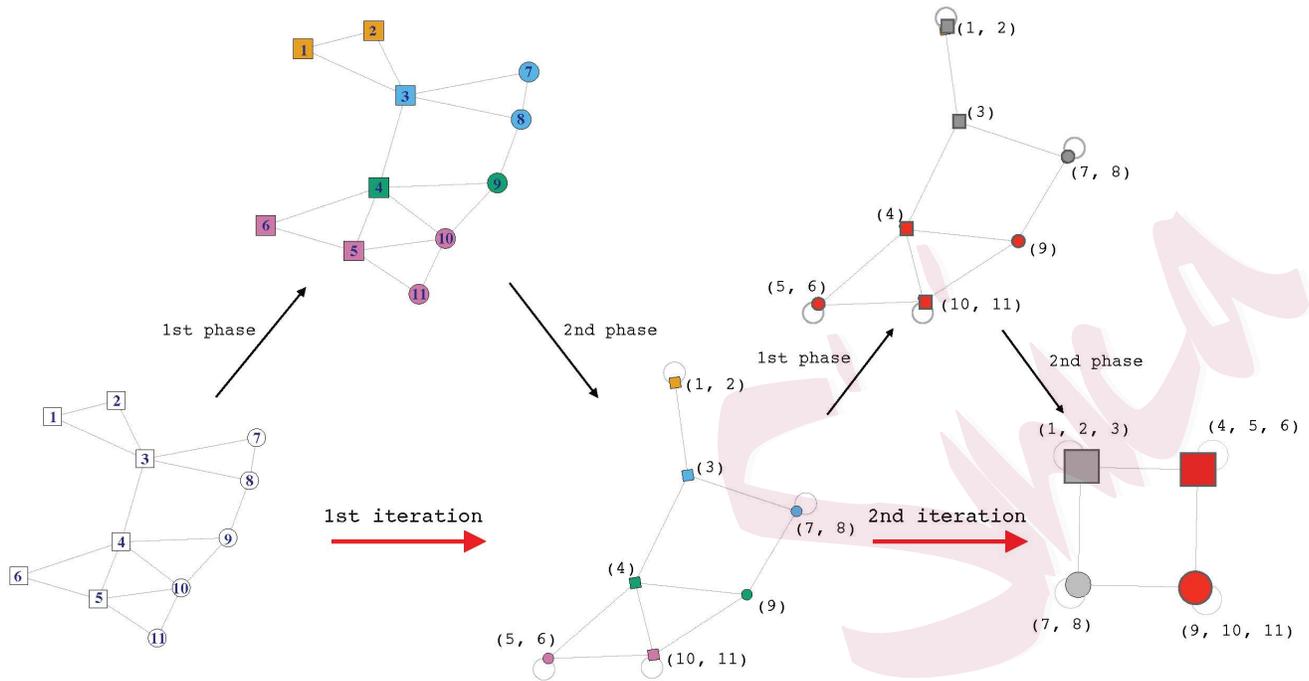


Figure 3.2: A visualization of the steps in the proposed algorithm. The two types of nodes are represented by squares and circles, respectively. Nodes of the same color are in the same community. After the first iteration and the second iteration, each node in the graph has a bracket next to it indicating the nodes from the original graph it contains.

The algorithm can be summarized as follows.

Algorithm 1. Take the modularity matrix M as input:

1. Assign each node to its own unit.
2. Assign each unit to its own community.
3. For each unit i , place it into the neighboring community that leads to the largest modularity increase. If no such move is possible, unit i remains in its current community.

4. *Apply Step 3 repeatedly to the units in the network until no units can be moved.*
5. *If the modularity is higher than the modularity from the previous iteration, then merge the nodes of the same type in each community; each community is treated as a unit and return to Step 2. If not, output the community assignment and the modularity value from the previous iteration.*

The result of the algorithm depends on the initial ordering of the nodes. In addition, in Step 3, each node is assigned to the community that leads to the largest modularity increase. If there are several communities that all lead to the largest increase, one community is randomly selected. Hence, the Louvain method may not arrive at the same result in successive runs. In our analysis, we apply the Louvain method κ times with random node orderings to find the maximum of the modularity function. In general, κ should increase with the size and the complexity of the network. In our simulation and real data analysis, we set $\kappa = 100$. We do not observe notable improvements in the maximized modularity function for $\kappa > 100$. However, other networks of comparable or larger sizes may benefit if larger values of κ are selected.

In the implementation of the Louvain method, the decision of whether and where to move a node can be computed in $O(1)$ time. Consequently, the complexity per iteration is $O(m)$, where m is the total number of edges in the network. An upper bound on the total running time of the algorithm can be calculated as $O(rm)$, where r is the total number of iterations. A trivial upper bound on r , which gives the worst case, is $O(m^2)$. While no nontrivial upper bound has been established on the number of iterations, the method needs only tens of iterations in practice to converge.

We note that the Louvain maximization method does not require the number of commu-

nities to be pre-specified. In cases where it is desirable to fix the number of communities at K^* in the procedure, the Louvain method can still be applied. Specifically, if K^* is reached during the iterations in the algorithm, we would stop the procedure and output the community assignment. If K^* is not reached after the algorithm finishes, i.e., the algorithm finds $K > K^*$, then we would continue with the algorithm and stop once K^* is reached; when continuing the algorithm, a small modification is that in Step 3, we would move unit i into the neighboring community that leads to the smallest modularity decrease. Recently, a few data driven approaches for estimating the number of communities have been proposed, including the spectral method in Le and Levina (2015), the penalized likelihood method in Wang and Bickel (2017), the hypothesis testing method in Karwa et al. (2016) and the cross-validation approach in Li et al. (2016).

Note that bipartite networks are special cases of heterogeneous networks considered in our work (with $L = 2$, $A^{[1]} = \mathbf{0}$ and $A^{[2]} = \mathbf{0}$). Hence, our proposed method can be used to find communities in bipartite networks (also referred to as bi-clustering). Many work have investigated community detection in bipartite networks. For example, Barber (2007) and Pesantez and Kalyanaraman (2017) considered a modularity based approach and used spectral methods for maximization, Larremore et al. (2014) considered a stochastic blockmodel approach and Rohe et al. (2016) considered a spectral clustering approach. Compared to the existing methods, our approach can be applied to very large bipartite networks and the number of communities K does not need to pre-specified.

4. Consistency

The consistency of community detection approaches for homogeneous networks has been studied extensively (Bickel and Chen, 2009; Rohe et al., 2011; Zhao et al., 2012; Jin, 2015).

However, theoretical properties of community detection methods for heterogeneous networks are largely unaddressed. In this section, we investigate the consistency property of our proposed method under a heterogeneous stochastic blockmodel framework. Consistency of community detection is generally investigated under a stochastic blockmodel framework (Rohe et al., 2011; Zhao et al., 2012; Abbe and Sandon, 2015; Zhang and Chen, 2016; Abbe, 2017; Vu, 2018). The consistency property of our method when applied to bipartite networks or multipartite networks follow as special cases.

Consider a heterogeneous network $\mathcal{G} = (\bigcup_{i=1}^L V^{[i]}, \mathcal{E} \cup \mathcal{E}^+)$ with latent community labels $\mathbf{c}^{[l]} = (c_1^{[l]}, \dots, c_{n_l}^{[l]})$, $l = 1, \dots, L$, where $c_i^{[l]} \in \{1, \dots, K\}$ is the community that the i th node of type- $[l]$ belongs to. Write $\mathcal{C} = (\mathbf{c}^{[1]}, \dots, \mathbf{c}^{[L]})$ and $n = \sum_{l=1}^L n_l$. We assume that the sizes of $V^{[l]}$, $l = 1, \dots, L$, are balanced, i.e., $\min_l n_l/n$ is bounded away from zero. We define a community detection criterion $F(\mathcal{C}, \mathcal{G})$ to be consistent if

$$\hat{\mathcal{C}} = \arg \max_{\mathcal{C}} F(\mathcal{C}, \mathcal{G})$$

satisfies

$$\forall \epsilon > 0, \quad P \left[\frac{1}{n} \sum_{l=1}^L \sum_{i=1}^{n_l} I(\hat{c}_i^{[l]} \neq c_i^{[l]}) < \epsilon \right] \rightarrow 1 \text{ as } n \rightarrow \infty.$$

This definition of consistency is a generalization of the one proposed in Zhao et al. (2012) for homogeneous networks. The definition requires that the error rate tends to zero in probability as the number of nodes goes to infinity.

Next we introduce the heterogeneous stochastic blockmodel, which serves as the framework of our theoretical development. Consider a heterogeneous network $\mathcal{G} = (\bigcup_{i=1}^L V^{[i]}, \mathcal{E} \cup \mathcal{E}^+)$ with latent community label \mathcal{C} . Write the adjacency matrix of $G_l(V^{[l]}, E^{[l]})$ as $A^{[l]}$, $l = 1, \dots, L$, and the bi-adjacency matrix of $G_{l_1 l_2}(V^{[l_1]} \cup V^{[l_2]}, E^{[l_1 l_2]})$ as $A_{ij}^{[l_1 l_2]}$, $1 \leq l_1 \neq l_2 \leq L$. In a heterogeneous stochastic blockmodel, each $A_{ij}^{[l]}$ is an independent Bernoulli random

variable with

$$E(A_{ij}^{[l]} \mid c_i^{[l]} = a, c_j^{[l]} = b) = P_{ab}^{[l]},$$

and each $A_{ij}^{[l_1 l_2]}$ is an independent Bernoulli random variable with

$$E(A_{ij}^{[l_1 l_2]} \mid c_i^{[l_1]} = a, c_j^{[l_2]} = b) = P_{ab}^{[l_1 l_2]},$$

where $P^{[l]}$ is a symmetric $K \times K$ probability matrix specifying the connecting probabilities between different communities of type- $[l]$ nodes, and $P^{[l_1 l_2]}$ is a $K \times K$ probability matrix specifying the connecting probabilities between type- $[l_1]$ nodes and type- $[l_2]$ nodes in different communities. Note that by definition, we have $P^{[l_1 l_2]} = P^{[l_2 l_1]}'$. Define $\pi^{[l]} = (\pi_1^{[l]}, \dots, \pi_K^{[l]})$ where $\pi_k^{[l]} = \frac{1}{n} \sum_{i=1}^{n_l} I(c_i^{[l]} = k)$, $l = 1, \dots, L$.

To ensure sparsity, the entries in the probability matrices need to tend to zero as the network grows in size. Otherwise, the network is going to become unrealistically dense. Following Bickel and Chen (2009), we define the expected degree $\lambda_n = n\rho_n$, where $\rho_n \equiv P(\text{Edge}) \rightarrow 0$. We can reparameterize $P^{[l]}$ as $\tilde{P}^{[l]} = \rho_n P^{[l]}$, where $P^{[l]}$ is fixed as $n \rightarrow \infty$. This reparameterization allows us to separate ρ_n from the structure of the network. See Bickel and Chen (2009) for a more detailed discussion of the reparameterization.

Consider the modularity function $Q(\mathcal{B}, \mathcal{G})$ in (2.8). The assignment matrix \mathcal{B} and the assignment vector $\mathcal{E} = (\mathbf{e}^{[1]}, \dots, \mathbf{e}^{[L]})$ with $\mathbf{e}^{[l]} = (e_1^{[l]}, \dots, e_{n_l}^{[l]})$, $l = 1, \dots, L$, have one to one correspondence. To simplify the notations, we write the modularity function $Q(\mathcal{B}, \mathcal{G})$ as $Q'(\mathcal{E}, \mathcal{G})$ in this section. The consistency property of the proposed heterogeneous network community detection criterion $Q'(\mathcal{E}, \mathcal{G})$ is introduced in the following theorem.

Theorem 2. Consider $\mathcal{G}(\bigcup_{i=1}^L V^{[i]}, \mathcal{E} \cup \mathcal{E}^+)$ from a heterogeneous stochastic blockmodel with

parameters $P^{[l]}$ and $P^{[l_1 l_2]}$, $l = 1, \dots, L$, $1 \leq l_1 \neq l_2 \leq L$. Define

$$T_{ab}^{[l]} = \frac{\pi_a^{[l]} \pi_b^{[l]} P_{ab}^{[l]}}{\sum_{ab} \pi_a^{[l]} \pi_b^{[l]} P_{ab}^{[l]}}, \quad \text{and} \quad T_{ab}^{[l_1 l_2]} = \frac{\pi_a^{[l_1]} \pi_b^{[l_2]} P_{ab}^{[l_1 l_2]}}{\sum_{ab} \pi_a^{[l_1]} \pi_b^{[l_2]} P_{ab}^{[l_1 l_2]}}.$$

Write $W^{[l]} = T^{[l]} - (T^{[l]} \mathbf{1})(T^{[l]} \mathbf{1})'$ and $W^{[l_1 l_2]} = T^{[l_1 l_2]} - (T^{[l_1 l_2]} \mathbf{1})(T^{[l_1 l_2]} \mathbf{1})'$. If the parameters satisfy

$$\sum_{l=1}^L W_{aa}^{[l]} + \sum_{l_1 \neq l_2}^L W_{aa}^{[l_1 l_2]} > 0, \quad \sum_{l=1}^L W_{ab}^{[l]} + \sum_{l_1 \neq l_2}^L W_{ab}^{[l_1 l_2]} < 0 \quad \text{for all } a \neq b, \quad (4.1)$$

then the proposed modularity function $Q'(\mathcal{E}, \mathcal{G})$ is consistent as $\lambda_n \rightarrow \infty$.

We refer to the online supplementary material for the proof. This result on consistency suggests that if networks are from a heterogeneous stochastic blockmodel with K communities, the community labels obtained from maximizing the modularity function $Q'(\mathcal{E}, \mathcal{G})$ will approach the true community labels as the number of nodes goes to infinity. The consistency properties of modularity functions formulated for bipartite or multipartite networks follow as special cases of Theorem 2. This fills an existing gap in the literature of modularity function based network community detection. Consistency of bipartite network community detection has been investigated in a recent work by Rohe et al. (2016); in this work, the upper bound of misclassification rate is derived for spectral clustering based approaches.

Conditions (4.1) in Theorem 2 essentially require that, on average, edges are more likely to be established within communities than between communities, even though community structures may not exist in all different types of edges. One example is the parameters in Simulation Setting 3 (see Section 5). In that case, edges within type-[1] or type-[2] nodes have no community structure, but edges linking type-[1] nodes and type-[2] nodes have community structure. Note that this type of assortative condition, i.e., more edges within communities than between communities, is often required for algorithm based community

detection methods, such as modularity maximization and minimum-cut. For probabilistic model based approaches such as the stochastic block model, community is defined based on nodes who are *stochastically equivalent* (Fienberg et al., 1985). Two nodes are stochastically equivalent if the probability of any event pertaining to the graph remains unchanged if the two nodes are exchanged. In model based approaches, assortative conditions are generally not required to identify communities.

In the case when $L = 2$ and $K = 2$, the conditions in (4.1) are satisfied if

$$\begin{aligned} P_{11}^{[1]} + P_{11}^{[2]} + P_{11}^{[12]} + P_{11}^{[21]} &> P_{12}^{[1]} + P_{12}^{[2]} + P_{12}^{[12]} + P_{12}^{[21]} \\ P_{22}^{[1]} + P_{22}^{[2]} + P_{22}^{[12]} + P_{22}^{[21]} &> P_{12}^{[1]} + P_{12}^{[2]} + P_{12}^{[12]} + P_{12}^{[21]}. \end{aligned}$$

These conditions describe that, on average, edges are more likely to be established within communities.

In a bipartite graph with K communities, we have $L = 2$, $P_{ab}^{[1]} = 0$ and $P_{ab}^{[2]} = 0$, $1 \leq a \leq b \leq K$. Define

$$T_{ab}^{[12]} = \frac{\pi_a^{[1]} \pi_b^{[2]} P_{ab}^{[12]}}{\sum_{ab} \pi_a^{[1]} \pi_b^{[2]} P_{ab}^{[12]}}, \quad 1 \leq a \neq b \leq K.$$

Write $W^{[12]} = T^{[12]} - (T^{[12]} \mathbf{1})(T^{[12]} \mathbf{1})'$. In this case, the assortative mixing condition (4.1) would be simplified to

$$W_{aa}^{[12]} > 0, \quad W_{ab}^{[12]} < 0 \quad \text{for all } 1 \leq a \neq b \leq K. \quad (4.2)$$

In the simple case with $K = 2$, the conditions in (4.2) can be simplified to

$$P_{11}^{[12]} P_{22}^{[12]} > (P_{12}^{[12]})^2, \quad P_{11}^{[12]} P_{22}^{[12]} > (P_{21}^{[12]})^2,$$

which is satisfied if $P_{11}^{[12]} > P_{12}^{[12]}$, $P_{11}^{[12]} > P_{21}^{[12]}$, $P_{22}^{[12]} > P_{12}^{[12]}$ and $P_{22}^{[12]} > P_{21}^{[12]}$; these conditions describe settings in which edges are more likely to be established within communities than between communities.

Consider the homogeneous method in which the heterogeneous network with L types of nodes is divided into L homogeneous networks. In the l -th homogeneous network, to correctly label the type- $[l]$ nodes with K communities, a sufficient condition (Zhao et al., 2012) is

$$W_{aa}^{[l]} > 0, \quad W_{ab}^{[l]} < 0 \quad \text{for all } a \neq b, \quad (4.3)$$

where $T_{ab}^{[l]} = \frac{\pi_a^{[l]} \pi_b^{[l]} P_{ab}^{[l]}}{\sum_{ab} \pi_a^{[l]} \pi_b^{[l]} P_{ab}^{[l]}}$, $W^{[l]} = T^{[l]} - (T^{[l]} \mathbf{1})(T^{[l]} \mathbf{1})'$. Hence, to achieve the consistency result in Theorem 2, one would need $W_{aa}^{[l]} > 0$, $W_{ab}^{[l]} < 0$ for all $l = 1, \dots, L$. In comparison, Condition (4.1) only requires $\sum_{l=1}^L W_{aa}^{[l]} + \sum_{l_1 \neq l_2}^L W_{aa}^{[l_1 l_2]} > 0$ and $\sum_{l=1}^L W_{ab}^{[l]} + \sum_{l_1 \neq l_2}^L W_{ab}^{[l_1 l_2]} < 0$ for all $a \neq b$. Essentially, (4.3) requires that community structure exists in each type of edges.

5. Simulation Study

In this section, we evaluate the performance of the proposed method through simulated heterogeneous networks, and compare it to the performances of the following methods:

- **Method 1 (homogeneous method)**: treat the whole heterogeneous network as one homogeneous network, i.e., do not distinguish the different types of nodes and edges;
- **Method 2 (homogeneous method)**: decompose the heterogeneous network with L different types of nodes into L homogeneous networks and consider each homogeneous network separately, i.e., discard information from the edges linking different types of nodes.

The community assignments from Method 1 and Method 2 are obtained through maximizing the modularity functions defined on the homogeneous networks (Newman and Girvan, 2004).

When finding communities using our proposed method, Method 1 and Method 2, we do not fix the number of the communities and treat it as an unknown quantity.

The model used to generate heterogeneous networks has two types of nodes ($L = 2$) and three communities ($K = 3$). We consider a stochastic block model type of structure with the probability matrix given as

$$P = \begin{pmatrix} P^{[1]} & P^{[12]} \\ P^{[21]} & P^{[2]} \end{pmatrix},$$

where

$$\begin{aligned} P^{[1]} &= p_1 \mathbf{1}_K \mathbf{1}'_K + r_1 \mathbf{I}_K, \\ P^{[2]} &= p_2 \mathbf{1}_K \mathbf{1}'_K + r_2 \mathbf{I}_K, \\ P^{[12]} = P^{[21]} &= p_3 \mathbf{1}_K \mathbf{1}'_K + r_3 \mathbf{I}_K, \end{aligned}$$

where $\mathbf{1}_K$ is the K -vector of 1's and \mathbf{I}_K is the K -by- K identity matrix. Here $P^{[1]}$ is a 3×3 probability matrix characterizing the connecting probabilities between type-[1] nodes in the three communities. For example, $P_{22}^{[1]}$ is the probability that there is an edge between two type-[1] nodes that are both in the second community. Similarly, $P^{[2]}$ is the probability matrix characterizing the connecting probabilities between type-[2] nodes in the three communities; $P^{[12]}$ and $P^{[21]}$ are the probability matrices characterizing the connecting probabilities between nodes of different types in the three communities. In the type-[1] (type-[2]) homogeneous network, p_1 (p_2) represents the inter-community connecting probability and $p_1 + r_1$ ($p_2 + r_2$) represents the intra-community connecting probability. In the type-[1] to type-[2] bipartite network, p_3 describes the inter-community connecting probability and $p_3 + r_3$ describes the intra-community connecting probability. The strength of the community structure is therefore regulated by r_1 , r_2 and r_3 .

Our main goal in this simulation study is to study how the clustering results from the proposed method, Method 1 and Method 2 change with r_3 under different settings. A higher

value for r_3 results in a stronger intra-community connection between a type-[1] node and a type-[2] node, i.e., more information is contained in the edges linking different types of nodes.

We consider three simulation settings in this section. In all three settings, we gradually increase r_3 and compare the performances of the proposed method, Method 1 and Method 2. In Simulation 1, the two homogeneous networks of type-[1] nodes and type-[2] nodes both have weak community structures. In Simulation 2, the homogeneous network of type-[1] nodes has weak community structure and the homogeneous network of type-[2] nodes has no community structure. In Simulation 3, neither of the two homogeneous networks has a community structure. We set the number of type-[1] nodes to 600 and assign 200 nodes to each community; we set the number of type-[2] nodes to 300 and assign 100 nodes to each community.

Before we discuss the results from the simulations, we first introduce a numerical measure to quantify the difference between two partitions. In this work, we adopt the normalized mutual information measure (Danon et al., 2005). Consider the community assignment $\{x_i\}$ and $\{y_i\}$, where x_i and y_i indicate the cluster labels of vertex i in partitions \mathcal{X} and \mathcal{Y} , respectively. Assume that labels x and y are the observed values of two random variables X and Y . The normalized mutual information (NMI) is measured by

$$NMI(\mathcal{X}, \mathcal{Y}) = \frac{2I(X, Y)}{H(X) + H(Y)},$$

where $I(X, Y) = H(X) - H(X|Y)$ is the mutual information and $H(X) = -\sum_x P(x) \log P(x)$ is the Shannon entropy of X . The normalized mutual information equals 1 if the two partitions are identical, and its expected value is 0 if the two partitions are independent.

Simulation Setting 1:

In this simulation, we set the parameters $p_1 = 0.1$, $r_1 = 0.05$, $p_2 = 0.2$, $r_2 = 0.1$ and $p_3 = 0.05$. Under this setting, there are weak community structures within both type-[1] nodes and type-[2] nodes. We can see type-[1] nodes and type-[2] nodes behave quite differently; compared to type-[1] nodes, type-[2] nodes are much more densely connected amongst themselves. In this simulation, we gradually change r_3 from 0.05 to 0.15. For each r_3 value, we simulate 100 heterogeneous networks from the model. For each heterogeneous network, we apply the proposed method, Method 1, Method 2 and calculate the NMI between the obtained community detection results and the true community membership. The average of the NMI from the 100 simulations is summarized in the top panel of Figure 5.3.

We can see that the proposed method performs better than Method 1 and Method 2 on all values of r_3 . Method 2 does not have satisfactory performance, with average NMI below 0.25 for both types of nodes. This is because the two homogeneous networks of type-[1] nodes and type-[2] nodes both have very weak community structures, and Method 2 does not take into consideration the edges linking different types of nodes. Note that our proposed method has good performance even when the connections between type-[1] nodes and type-[2] nodes display a weak community structure with $r_3 = 0.05$.

Simulation Setting 2:

In this simulation, we set the parameters $p_1 = 0.1$, $r_1 = 0.05$, $p_2 = 0.2$, $r_2 = 0$, $p_3 = 0.05$. Under this setting, the homogeneous network of type-[2] nodes has no community structure. Similar to Simulation 1, we gradually increase r_3 from 0.05 to 0.15 and simulate 100 heterogeneous networks from the model. The average of the NMI from the 100 simulations is summarized in the middle panel of Figure 5.3.

We can see the proposed method outperforms Method 1 and Method 2 on all values of r_3 .

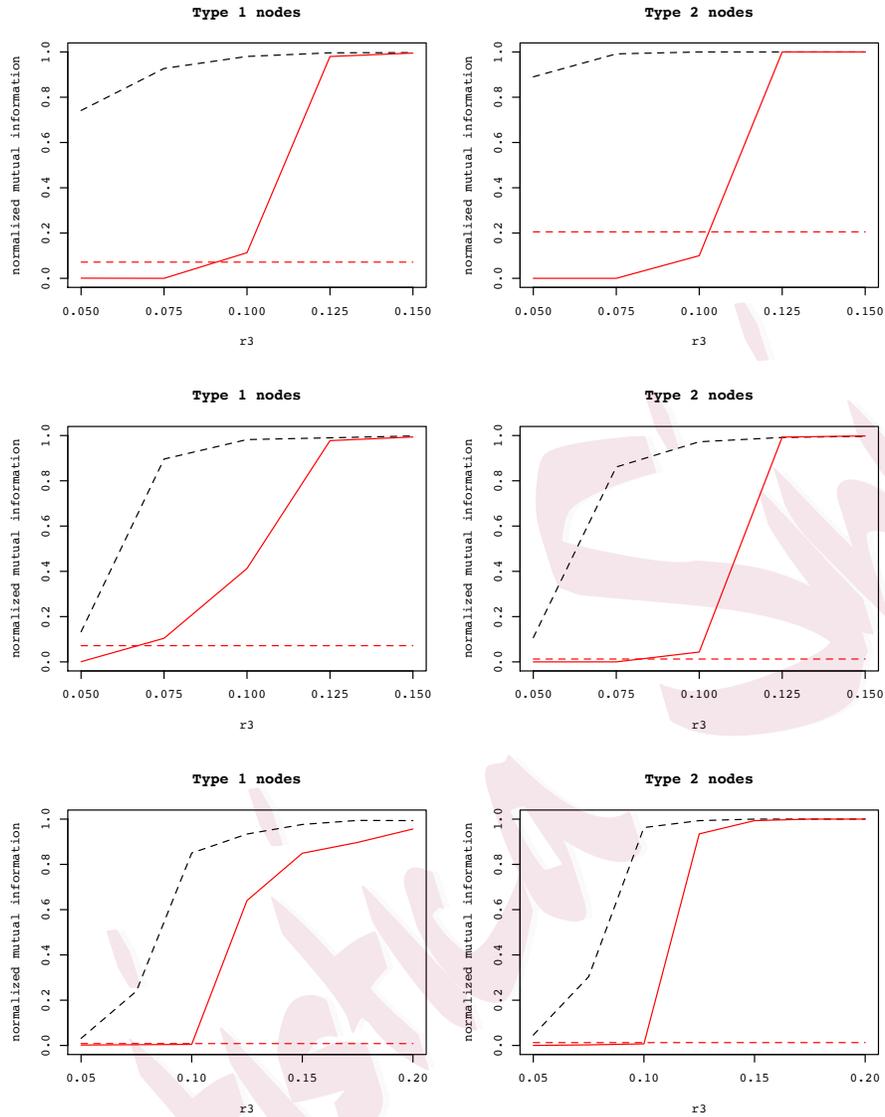


Figure 5.3: Average NMI between the true community membership and the community membership obtained from the proposed method (black dashed line), Method 1 (red solid line) and Method 2 (red dashed line). Top panel: results from Simulation 1; middle panel: results from Simulation 2; bottom panel: results from Simulation 3.

For type-[2] nodes, the NMI from Method 2 is 0 since $r_2 = 0$. When $r_3 = 0.05$, the proposed method yields unsatisfactory performance. This is because the community structure is very weak within type-[1] nodes, and between type-[1] and type-[2] nodes. When r_3 increases

slightly to 0.075, we see notable improvement in the performance of the proposed method.

Simulation Setting 3:

In this simulation, we set the parameters $p_1 = 0.1$, $r_1 = 0$, $p_2 = 0.2$, $r_2 = 0$, $p_3 = 0.05$. Under this setting, there are no community structures within type-[1] nodes or type-[2] nodes. In this simulation, we gradually increase r_3 from 0.05 to 0.20. The average of the NMI from 100 simulations between the true membership and the community membership calculated from the proposed method, Method 1 and Method 2 are summarized in the bottom panel of Figure 5.3.

We can see the proposed method has the best performance out of the three methods consistently. The NMI from Method 2 is 0 for both type-[1] nodes and type-[2] nodes, since there are no community structures within type-[1] nodes or within type-[2] nodes. For $r_3 = 0.05$ and 0.075, the proposed method yields NMI below 0.4 for both types of nodes. The low NMI is a result of the weak community structure in the simulated heterogeneous networks, with both r_1 and r_2 equal 0. When r_3 increases to 0.1, we see a significant improvement in the performance of the proposed method, while Method 1 still performs poorly.

6. Real Data Application

6.1. DBLP Dataset

DBLP (Digital Bibliography & Library Project) is a computer science bibliography website, listing more than 3.4 million journal articles, conference papers, and other publications in computer science. Gao et al. (2009) and Ji et al. (2010) extracted a connected subset of the DBLP dataset, containing bibliographical records from four research areas: database, data mining, information retrieval, and artificial intelligence. This network contains three

types of nodes: paper, conference, and author. Among the three types of nodes, there are two types of edges: paper-conference (paper published at conference), paper-author (paper written by author). This dataset consists of 14,376 papers written by 14,475 authors, and published at 20 conferences. Each one of the 20 conferences is labeled with the research area it covers. Each research area has five conferences. The true research area is available for 4,057 authors that are connected to a subset of 14,328 papers, covering all 20 conferences. The objective in this real data application is to correctly identify the research areas of the authors. Since error rates can be calculated only for labeled authors, we focus our data analysis on this labeled subset of the data.

Applying the proposed maximization method to the heterogeneous network modularity function with $K = 4$ and $\kappa = 100$, we cluster the heterogeneous network into four communities with the maximized modularity value 0.65. One application of the proposed maximization procedure takes less than 20 seconds on an iMac with 3.2 GHz Intel Core i5. We label the research area of each community using the conferences each community contains (see Table 6.1). The misclassification rate for the conferences is 0%. We label the authors in each community with the research area that the community belongs to and compare the labels to the ground truth. The misclassification rate for the authors is 8.84%.

Community	Conferences	Research Area
1	PODS, ICDE, SIGMOD, EDBT, VLDB	Database
2	ICDM, PAKDD, PKDD, KDD, SDM	Data Mining
3	AAAI, IJCAI, ECML, ICML, CVPR	Artificial Intelligence
4	WWW, WSDM, CIKM, ECIR, SIGIR	Information Retrieval

Table 6.1: The conferences in each community and the research areas the conferences cover.

We also considered Method 1 and Method 2 described in Section 5. Method 1 considers a homogeneous model in which the heterogeneous network is treated as a homogeneous network, i.e., do not distinguish the different types of nodes and edges. To find the community structure in the homogeneous model, we used the standard Newman-Girvan modularity function and the Louvain maximization approach with K fixed at ground truth, i.e., $K = 4$. The identified communities are very difficult to interpret. For example, one community contains only papers and one community contains only conferences. This is not surprising since the homogeneous method treats author nodes, paper nodes and conference nodes equally, even though they behave differently in the DBLP network. In comparison, our proposed heterogeneous model clusters authors, papers and conferences into the four known research area, with misclassification rates 0% and 8.84% for conferences and authors, respectively. Method 2 cannot be applied since there are no author-author, paper-paper or conference-conference connections.

6.2. MovieLens Dataset

MovieLens (<https://movielens.org/>) is a website that allows users to review movies; based on their reviews, users can receive personalized movie recommendations. The website was created in 1997 by a research lab in the Department of Computer Science and Engineering at the University of Minnesota to collect research data (Harper and Konstan, 2015). The MovieLens dataset (<https://grouplens.org/datasets/movielens/>) contains reviews from 943 users on 1,682 movies from 18 movie genre including action, adventure, animation, children's, comedy, crime, documentary, drama, fantasy, film-noir, horror, musical, mystery, romance, sci-fi, thriller, war and western. Using the MovieLens dataset, we construct a heterogeneous network with three types of node: user, movie and genre, and two types of edges: user-movie

(movie reviewed by user), movie-genre (movie contained in genre). The objective in this real data application is to find communities in this heterogeneous network. The identified communities can be used to classify movies, users, and make movie recommendations since users in the same community are more likely to watch the movies contained in this community.

Applying the proposed heterogeneous network community detection technique with $\kappa = 100$, we identified 7 communities with maximized modularity value 0.33. Table 6.2 shows the genre node(s), number of movies and users (percentage of the total) contained in each community. We can see Community 2 is the most popular community among users (contains about 36% of total users) and Community 7 is the least popular community among users (contains less than 2% of the total users). An interesting observation is that each community contains a distinctive type of movies. This observation can in turn help us understand the movie preferences of users contained in each community. For example, users in Community 6 like movies from the animation, children's, fantasy and musical genre. This preference is very different from that of users in Community 2 who like movies from the crime, file-noir, mystery and thriller genre. We can also see horror and documentary each form its own small community.

In the MovieLens dataset, demographic information such as gender and occupation is available for some of the users. Over 70% of the identified male users are in Communities 1, 2 and 4; over 60% of the identified female users are in Communities 1 and 2. We find that Communities 3 and 7 are the least popular among users who are listed as students (the two communities together contain less than 4% of the student users). Communities 1 and 2 are the most popular among users who are listed as educators or administrators (the two communities together contain over 70% of the educator users and 55% of the administrator

Community	Movie Genre	# of movies (% of total)	# of users (% of total)
1	Drama, War	27%	21%
2	Crime, Film-Noir, Mystery, Thriller	16%	36%
3	Horror	5%	3%
4	Action, Adventure, Sci-Fi, Western	14%	17%
5	Comedy, Romance	26%	16%
6	Animation, Children’s, Fantasy, Muscial	9%	6%
7	Documentary	3%	1%

Table 6.2: Movie genre, number of movies and users contained in the identified communities.

users). Community 4 is the most popular among users who are listed as programmers or engineers (this community contains over 30% of the engineer users and over 30% of the programmer users).

We also considered the same homogeneous model (Method 1) as in Section 6.1. In this dataset, we do not know the ground truth; therefore, the number of communities K is not fixed. The homogeneous method found $K = 5$ clusters. However, the result is difficult to interpret. For example, one cluster contains only movies but no users and genres. One cluster contains only movies and users but no movie genres. This is not entirely surprising as the homogeneous method treats users, movies and movie genres equally, even though they behave differently in the heterogeneous network.

7. Discussion

In this paper, we propose a modularity based framework for community detection on

heterogeneous networks. Specifically, we define a null model for heterogeneous networks. Furthermore, we propose a modularity maximization method that can handle very large networks. We show that under a heterogeneous stochastic blockmodel, the proposed modularity function is consistent as a community detection criterion. The proposed community detection approach performs well with both simulated and real-world networks.

Regarding other heterogeneous network based methods, a spectral clustering approach was proposed in Sengupta and Chen (2015). In spectral clustering methods, it is assumed that the number of communities K is known *a priori*, whereas in our method, we do not require K to be pre-specified. Moreover, the heterogeneous spectral clustering method clusters each type of nodes into K homogeneous clusters, whereas our method cluster all nodes into K heterogeneous clusters. Therefore, these two methods are not directly comparable. For finite sample performance of modularity based method and spectral clustering based method for network community detection, please see Yang et al. (2016) and references therein.

If we treat the heterogeneous network with L types of nodes and K communities as a homogeneous network with $L \times K$ communities, modularity based community detection would require all $L \times K$ communities to be assortative, i.e., more edges within communities and less edges between communities (Zhao et al., 2012; Zhang and Chen, 2016). This is in fact a much stronger condition than that needed for heterogeneous community detection, which only requires on average, more edges are placed within the K heterogeneous communities and less edges are placed between these communities. See additional simulation results in the supplementary material.

We note that the maximization of the proposed modularity function is not tied to the Louvain method. In fact, several existing modularity maximization techniques can be applied

to our setting with some modifications, such as the spectral method based on the eigen decomposition of the modularity matrix (Newman, 2006) or the stochastic maximization method (Massen and Doye, 2005). However, in practice, we find that the Louvain method yields better modularity maximum than the other methods and is computationally more efficient. Another suitable approach is to apply the spectral method proposed in Sengupta and Chen (2015), which performs a K-means clustering of the K eigenvectors corresponding to the K largest eigenvalues of the regularized graph Laplacian matrix.

It is important to point out that in our work, we are interested in finding K assortative communities that are heterogeneous, i.e., communities with L different types of nodes. However, if one is interested in finding K communities such that each community is homogeneous and that the communities are assortative, i.e., more links within communities and less links between communities. This becomes a different community detection problem, as the motivation and interpretations of the communities under this setting would be very different from the one considered in our work.

The proposed method can be extended to directed heterogeneous networks. Several approaches have been proposed for finding communities in directed homogeneous networks using modified modularity functions (see Fortunato, 2010 for a review). To incorporate directed edges into our framework, we need to define a null model for directed heterogeneous networks. Furthermore, we need to calculate the expectations under the null model. This is an interesting topic to investigate.

Supplementary Material

The online supplementary material includes proofs for Theorem 1, Theorem 2 and additional simulation results.

Acknowledgements

This work was supported in part by National Science Foundation grant DMS-1406455.

References

- Abbe, E., and Sandon, C. (2015). Detection in the stochastic block model with multiple clusters: proof of the achievability conjectures, acyclic BP, and the information-computation gap. *arXiv preprint arXiv:1512.09080*.
- Abbe, E. (2017). Community detection and stochastic block models: recent developments. *arXiv preprint arXiv:1703.10146*.
- Agrawal, G., and Kempe, D. (2008). Modularity-maximizing graph communities via mathematical programming. *The European Physics Journal B* **66**, 409-418.
- Airoldi, E. M., Blei, D. M., Fienberg, S. E., and Xing, E. P. (2008). Mixed-membership stochastic blockmodels. *Journal of Machine Learning Research* **9**, 1981-2014.
- Barber, M. J. (2007). Modularity and community detection in bipartite networks. *Physical Review E* **76(6)**, 066102.
- Bender, E. and Canfield, R. (1978), "The asymptotic number of labeled graphs with given degree sequences," *Journal of Combinatorial Theory A* **24**, 296-307.
- Bickel, P., and Chen, A. (2009). A non-parametric view of network models and Newman-Girvan and other modularities. *Proceedings of the National Academy of Sciences* **106**, 21068-21073.
- Blondel, V. D., Guillaume, J. L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* **10**, P10008.
- Blondel, V. D. (2011). The Louvain method for community detection in large networks. <https://perso.uclouvain.be/vincent.blondel/research/louvain.html>.

- Bollobás, B., and McKay, B. D. (1986). The number of matchings in random regular graphs and bipartite graphs. *Journal of Combinatorial Theory, Series B* **41(1)**, 80-91.
- Brandes, U., Delling, D., Gaertler, M., Gorke, R., Hoefer, M., Nikoloski, Z., and Wagner, D. (2008). On modularity clustering. *IEEE Transactions on Knowledge and Data Engineering* **20**, 172-188.
- Chung, F., and Lu, L. (2002). Connected components in random graphs with given expected degree sequences. *Annals of Combinatorics* **6(2)**, 125-145.
- Clauset, A., Newman, M. E. J., and Moore, C. (2004). Finding community structure in very large networks. *Physical Review E* **70**, 066111.
- Danon, L., Diaz-Guilera, A., Duch, J., and Arenas, A. (2005). Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment* **09**, P09008.
- Fienberg, S. E., Meyer, M. M. and Wasserman, S. S. (1985). Statistical Analysis of Multiple Sociometric Relations. *Journal of the American Statistical Association* **80**, 51-67.
- Fortunato, S. (2010). Community detection in graphs. *Physics Reports* **428**, 75-174.
- Gao, J., Liang, F., Fan, W., Sun, Y., and Han, J. (2009). Graph-based consensus maximization among multiple supervised and unsupervised models. *Advances in Neural Information Processing Systems* **22**, 585-593.
- Harper, F. M., and Konstan, J. A. (2016). The movielens datasets: history and context. *ACM Transactions on Interactive Intelligent Systems* **5(4)**, 19.
- Ji, M., Sun, Y., Danilevsky, M., Han, J., and Gao, J. (2010). Graph regularized transductive classification on heterogeneous information networks. *Proceeding of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 570-586.
- Jin, J. (2015). Fast community detection by SCORE. *The Annals of Statistics* **43**, 57-89.
- Karwa, V., Pati, D., Petrovi, S., Solus, L., Alexeev, N., Raic, M., Wilburne D., Williams R. and Yan, B. (2016). Exact tests for stochastic block models. *arXiv preprint*

arXiv:1612.06040.

Larremore, D. B., Clauset, A., and Jacobs, A. Z. (2014). Efficiently inferring community structure in bipartite networks. *Physical Review E* **90(1)**, 012805.

Le, C. M., and Levina, E. (2015). Estimating the number of communities in networks by spectral methods. *arXiv preprint arXiv:1507.00827*.

Li, T., Levina, E., and Zhu, J. (2016). Network cross-validation by edge sampling. *arXiv preprint arXiv:1612.04717*.

Liu, X., Liu, W., Murata, T., and Wakita, K. (2014). A framework for community detection in heterogeneous multi-relational networks. *Advances in Complex Systems* **17(06)**, 1450018.

Massen, C., and Doye, J. (2005). Identifying communities within energy landscapes. *Physical Review E* **71**, 046101.

McKay, B. D. (2010). Subgraphs of random graphs with specified degrees. *Proceedings of the International Congress of Mathematicians 2010* **4**, 2489-2501.

Newman, M. E. J. (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical Review E* **74**, 035104.

Newman, M. E. J., and Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E* **69**, 026113.

Perry, P., and Wolfe, P. (2012). Null models for network data. *arXiv preprint arXiv:1201.5871*.

Pesantez, P. G., and Kalyanaraman, A. (2017). Efficient Detection of Communities in Biological Bipartite Networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

Reichardt, J., and Bornholdt, S. (2006). Statistical mechanics of community detection. *Physical Review E* **74**, 016110.

Rohe, K., Chatterjee, S., and Yu, B. (2011). Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics* **39(4)**, 1878-1915.

- Rohe, K., Qin, T., and Yu, B. (2016). Co-clustering directed graphs to discover asymmetries and directional communities. *Proceedings of the National Academy of Sciences* **113(45)**, 12679-12684.
- Sengupta, S., and Chen, Y. (2015). Spectral clustering in heterogeneous networks. *Statistica Sinica* **25**, 1081-1106.
- Shi, J., and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**, 888-905.
- Sun, Y., and Han, J. (2012). Mining heterogeneous information networks: principles and methodologies. *Synthesis Lectures on Data Mining and Knowledge Discovery* **3(2)**, 1-159.
- Vu, V. (2018). A simple SVD algorithm for finding hidden partitions. *Combinatorics, Probability and Computing* **27(1)**, 124-140.
- Wakita, K., and Tsurumi, T. (2007). Finding community structure in mega-scale social networks. *Proceedings of the 16th International Conference on World Wide Web*, 1275-1276.
- Wang, Y. R., and Bickel, P. J. (2017). Likelihood-based model selection for stochastic block models. *The Annals of Statistics* **45(2)**, 500-528.
- Zhang, J., and Cao, J. (2018). Finding common modules in a time-varying network with application to the *Drosophila Melanogaster* gene regulation network. *Journal of the American Statistical Association* **112(519)**, 994-1008.
- Zhang, J., and Chen, Y. (2016). A hypothesis testing framework for modularity based network community detection. *Statistica Sinica* **27**, 437-456.
- Zhao, Y., Levina, E., and Zhu, J. (2012). Consistency of community detection in networks under degree-corrected stochastic block models. *The Annals of Statistics* **40**, 2266-2292.

Department of Management Science, University of Miami, Coral Gables, FL 33124, USA.

E-mail: ezhang@bus.miami.edu

Department of Statistics, University of Illinois at Urbana-Champaign, Champaign, IL 61820,
USA.

E-mail: yuguo@illinois.edu

Statistica Sinica