International Journal of Modern Physics C
Vol. 26, No. 9 (2015) 1550096 (14 pages)
© World Scientific Publishing Company
DOI: 10.1142/S0129183115500965

Community detection in bipartite networks using weighted symmetric binary matrix factorization

Zhong-Yuan Zhang
School of Statistics and Mathematics
Central University of Finance and Economics, Beijing, P. R. China
zhyuanzh@gmail.com

Yong-Yeol Ahn
School of Informatics and Computing
Indiana University Bloomington, IN, USA
yyahn@indiana.edu

Received 8 September 2014
Accepted 4 December 2014
Published 26 January 2015

In this paper, we propose weighted symmetric binary matrix factorization (wSBMF) framework to detect overlapping communities in bipartite networks, which describes the relationships between two types of nodes. Our method improves performance by recognizing the distinction between two types of missing edges — ones among the nodes in each node type and the others between two node types. Our method can also explicitly assign community membership and distinguish outliers from overlapping nodes, as well as incorporating existing knowledge on the network. We propose a generalized partition density for bipartite networks as a quality function, which identifies the most appropriate number of communities. The experimental results on both synthetic and real-world networks demonstrate the effectiveness of our method.

Keywords: Bipartite network; weighted symmetric binary matrix factorization; partition density.

1. Introduction

Community structure is a common characteristic of various complex networks found in biological, social, and information systems, etc.[1–8] A community is commonly defined as a densely interconnected set of nodes that is loosely connected with the rest of the network.[1] Studies have shown that community structures are highly relevant to the organization and functions of the network. For instance, communities in social networks correspond to social circles[1]; communities in protein–protein interaction networks capture functional modules[5,3] and communities affect the spread of behaviors and ideas.[3,9,10]

Although numerous community detection methods have been proposed, relatively few methods are designed for bipartite networks.[11–17] A bipartite network $G(\Delta, \Gamma, E)$ contains two disjoint types of nodes, Δ and Γ, and the edge set E connecting the two parts. There is no edge among vertices in Δ and among those in Γ. Many systems can be naturally modeled as bipartite networks.[14,18] For instance, a metabolic network can be considered as a bipartite network of reactions and metabolites.[19] Many unipartite networks are derived from bipartite ones. For instance, a scientific collaboration network is derived from an author-paper bipartite network.[20] A community in a bipartite network $G(\Delta, \Gamma, E)$ can be defined as a set of nodes — from both Δ and Γ — that are densely interconnected. Bipartite community detection is not necessarily equivalent to unipartite community detection on the projected networks, because the projection often destroys important information.[12,14,21] Here, we would like to point out the difference between the missing edge among Δ and among Γ, and that between Δ and Γ. Imagine a network of people and their affiliations. With complete information about people's affiliation, the absence of edge $(i, j)(i \in \Delta, j \in \Gamma)$ means that the person i does not belong to the organization j. However, the absence of edge (i, k) $(i, k \in \Delta)$ simply indicates that we do not know the direct social relationships between i and k.

In our previous work, we proposed the Symmetric Binary Matrix Factorization (SBMF) to detect overlapping communities in unipartite networks and demonstrated its effectiveness.[22] In this paper, we propose weighted Symmetric Binary Matrix Factorization (wSBMF) model to detect overlapping communities in bipartite networks. The model can differentiate between the two kinds of missing edges in the bipartite network to improve detecting performance. The model allows us explicitly to assign community membership to nodes and distinguish outliers from overlapping nodes while providing a way to analyze the strength of membership and incorporate existing information. To quantify the goodness of the communities that we found, we generalize partition density and use it to select the most appropriate number of communities.

2. Methods

2.1. *Weighted symmetric binary matrix factorization*

The adjacency matrix of an undirected and unweighted simple graph G with n nodes can be defined as:

$$A_{ij} = \begin{cases} 1, & \text{if } i \sim j, \\ 0, & \text{if } i = j \text{ or } i \nsim j, \end{cases}$$

where $i \sim j$ means there is an edge and $i \nsim j$ means there is no edge.

Imagine an unweighted and undirected bipartite network $G(\Delta, \Gamma, E)$, which has n_Δ and n_Γ nodes in Δ and Γ, respectively, and an edge set E connecting the two parts. The corresponding adjacency matrix A can be split into four blocks after the

n_Δth row and the n_Δth column:

$$A = \begin{bmatrix} \mathbf{0}_{\boldsymbol{\Delta}} & B \\ B^T & \mathbf{0}_{\boldsymbol{\Gamma}} \end{bmatrix},$$

where $\mathbf{0}_{\boldsymbol{\Delta}}$ and $\mathbf{0}_{\boldsymbol{\Gamma}}$ are null matrices of size $n_\Delta \times n_\Delta$ and $n_\Gamma \times n_\Gamma$, respectively, and

$$B_{ij} = \begin{cases} 1, & \text{if } i \sim j, \quad i \in \Delta, j \in \Gamma, \\ 0, & \text{if } i \not\sim j, \quad i \in \Delta, j \in \Gamma. \end{cases}$$

The meaning of the zeros in $\mathbf{0}_{\boldsymbol{\Delta}}$, $\mathbf{0}_{\boldsymbol{\Gamma}}$ is different from that in B. If B captures all existing connections perfectly, then all zeros in B indicate the absence of the corresponding edges. By contrast, the zeros in $\mathbf{0}_{\boldsymbol{\Delta}}$ and $\mathbf{0}_{\boldsymbol{\Gamma}}$ represent missing information, rather than the absence of edges. To use this information, we introduce a weight matrix L of size $n \times n$ to handle these unobserved or missing values,[23] which can be defined as:

$$L_{ij} = \begin{cases} \gamma & \text{if } A_{ij} \text{ is observed,} \\ 0 & \text{if } A_{ij} \text{ is unobserved,} \end{cases}$$

where γ is a nonnegative weight parameter that captures the reliability of A_{ij}. For standard bipartite networks, L can be formulated as:

$$L = \begin{bmatrix} \mathbf{0}_{\boldsymbol{\Delta}} & \mathbf{I}_{\boldsymbol{\Delta},\boldsymbol{\Gamma}} \\ \mathbf{I}_{\boldsymbol{\Gamma},\boldsymbol{\Delta}} & \mathbf{0}_{\boldsymbol{\Gamma}} \end{bmatrix},$$

where $\mathbf{I}_{\boldsymbol{\Delta},\boldsymbol{\Gamma}}$ and $\mathbf{I}_{\boldsymbol{\Gamma},\boldsymbol{\Delta}}$ are matrices where all entries are one, meaning that only the zeros in B are considered. The sizes of $\mathbf{I}_{\boldsymbol{\Delta},\boldsymbol{\Gamma}}$ and $\mathbf{I}_{\boldsymbol{\Gamma},\boldsymbol{\Delta}}$ are $n_\Delta \times n_\Gamma$ and $n_\Gamma \times n_\Delta$, respectively.

Our wSBMF model can be defined as the following constrained nonlinear programming:

$$\begin{aligned} &\min_U \ \|L \circ (A - UU^T)\|_1 + \sum_i \left(1 - \Theta\left(\sum_j U_{ij}\right)\right) \\ &\text{subject to } U_{ij}^2 - U_{ij} = 0, \quad i = 1, 2, \ldots, n, \ j = 1, 2, \ldots, c, \end{aligned} \tag{1}$$

where $\circ$ represents element-wise multiplication (Hadamard product); A is the adjacency matrix of size $n \times n$ ($n = n_\Delta + n_\Gamma$); U is the community membership matrix such that $U_{it} = 1$ if node i is in the community t, and 0 if otherwise; Note that numerical experiments show that the Frobenius norm on the sparse adjacency matrix A often results in the ultra-sparsity of U, even null matrix U, which is not informative enough for real analysis. We use 1-norm instead to obtain more reasonable and explainable matrix U. 1-norm of a matrix X is the largest column sum of $\mathrm{abs}(X)$, where $\mathrm{abs}(X)_{ij} = \mathrm{abs}(X_{ij})$, and $\mathrm{abs}(\cdot)$ is the absolute value; Θ is the Heaviside step function such that for some matrix X,

$$\Theta(X)_{ij} := \begin{cases} 1 & \text{if } X_{ij} > 0; \\ 0 & \text{if } X_{ij} \le 0. \end{cases}$$

L chooses which entries of the adjacency matrix should be considered in the optimization and thus allows us to incorporate existing knowledge. For instance, if we already know that some edges are present between nodes in Δ, then we can update the corresponding elements of L from zero to γ. If we want to ignore edges in B, we can simply update the corresponding element of L from one to zero. We can even vary γ across elements if we can assess the reliability of the incorporated knowledge.

We initialize U by solving the following weighted Symmetric Nonnegative Matrix Factorization model:

$$\begin{aligned} &\min_{U} \ \|L \circ (A - UU^T)\|_F^2 \\ &\text{subject to} \quad U_{ij} \geq 0, \ i = 1, 2, \ldots, n, \ j = 1, 2, \ldots, c, \\ &\sum_{j=1}^{c} U_{ij} = 1, \quad i = 1, 2, \ldots, n. \end{aligned} \tag{2}$$

Then we fix U, and discretize the domain $\{u : 0 \leq u \leq \max(U)\}$ to find $\hat{u}$ that minimizes the following, simpler optimization problem:

$$\min_{U} \|L \circ (A - \Theta(U-u)\Theta(U-u)^T)\|_1 + \sum_{i} \left(1 - \sum_{j} \Theta(U-u)_{ij}\right), \tag{3}$$

where u is a scalar. Finally, we obtain the binary matrix U as follows:

$$U := \Theta(U - \hat{u}).$$

To optimize U for model (2), we initialize U using the algorithm of alternative least squares error developed for NMF[24,25]:

$$\begin{aligned} &\min_{U_1, U_2} \quad \|B - U_1 U_2^T\|_F^2 \\ &\text{subject to} \quad U_1 \geq 0, \ U_2 \geq 0. \end{aligned} \tag{4}$$

See Algorithm 1 Appendix.

Then, based on the boundedness theorem,[26–28] we normalize U_1 and U_2 to balance their scales:

$$U_1 = U_1 D_1^{-1/2} D_2^{1/2}, \quad U_2 = U_2 D_2^{-1/2} D_1^{1/2}, \tag{5}$$

where

$$\begin{aligned} D_1 &= \mathrm{diag}(\max U_1(:,1), \max U_1(:,2), \ldots, \max U_1(:,c)); \\ D_2 &= \mathrm{diag}(\max U_2(:,1), \max U_2(:,2), \ldots, \max U_2(:,c)); \end{aligned}$$

and $\mathrm{diag}(a_1, a_2, \ldots, a_n)$ is the diagonal matrix whose diagonal entries starting from the upper left corner are $a_1, a_2, \ldots, a_n$. $U_1(:,i)$ is the ith column of U_1. Finally, we merge U_1 and U_2 into U such that $U = \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}$, and employ the algorithm of multiplicative update rules for model (2). See Algorithm 2 Appendix.

2.2. *Model selection*

We have proposed a modified partition density to select the appropriate number of communities.[5,22] The modified partition density is defined as:

$$D = \sum_{\alpha=1}^{c} \frac{1}{q^{(\alpha)}} \frac{n^{(\alpha)}}{N} D^{(\alpha)},$$

where $D^{(\alpha)}$ is the partition density of community α:

$$D^{(\alpha)} = \frac{m^{(\alpha)} - \underline{m}^{(\alpha)}}{\overline{m}^{(\alpha)} - \underline{m}^{(\alpha)}},$$

and $\underline{m}^{(\alpha)} = (n^{(\alpha)} - 1)$, $\overline{m}^{(\alpha)} = n^{(\alpha)}(n^{(\alpha)} - 1)/2$ are the minimum and maximum possible numbers of links between the nodes in the community α, respectively; $n^{(\alpha)}$ and $m^{(\alpha)}$ are the number of nodes and the number of edges in the community α, respectively; $q^{(\alpha)} = \max_{j\in\alpha} l_j$ is the maximum number of community memberships (l_j) among the nodes (j) that belong to the community α; N is the sum of the sizes of different communities and the number of outliers.

Here, we generalize it for bipartite networks by transforming each bipartite community to a unipartite one and getting the corresponding partition density. For a community α, we define the subnetwork $G^{(\alpha)}$ as the set of nodes in α and the edges among them. The subnetwork has $n_\Delta^{(\alpha)}$ nodes in Δ and $n_\Gamma^{(\alpha)}$ nodes in Γ, and the corresponding adjacency matrix is

$$A^{(\alpha)} = \begin{bmatrix} \mathbf{0} & B^{(\alpha)} \\ B^{(\alpha)T} & \mathbf{0} \end{bmatrix}.$$

Then, we transform the bipartite subnetwork $G^{(\alpha)}$ to a unipartite subnetwork $G^{(\alpha)'}$ by overlaying the two projections onto Δ and Γ. The adjacency matrix $A^{(\alpha)}$ becomes:

$$A^{(\alpha)'} = \begin{bmatrix} B^{(\alpha)}B^{(\alpha)T} & B^{(\alpha)} \\ B^{(\alpha)T} & B^{(\alpha)T}B^{(\alpha)} \end{bmatrix},$$

and the diagonal elements indicate the number of neighbors in the other part that the corresponding node has. The values of $m^{(\alpha)}, \overline{m}^{(\alpha)}$ and $\underline{m}^{(\alpha)}$ are changed to:

$$m^{(\alpha)'} = \sum_{i,j} (A^{(\alpha)'} - \mathrm{diag}(A^{(\alpha)'}))_{ij}/2,$$

where $\mathrm{diag}(A^{(\alpha)'})$ is the diagonal matrix whose diagonal entries are those of $A^{(\alpha)'}$;

$$\overline{m}^{(\alpha)'} = \left[\frac{n_\Delta^{(\alpha)}(n_\Delta^{(\alpha)} - 1)}{2} n_\Gamma^{(\alpha)} + \frac{n_\Gamma^{(\alpha)}(n_\Gamma^{(\alpha)} - 1)}{2} n_\Delta^{(\alpha)} + n_\Delta^{(\alpha)} n_\Gamma^{(\alpha)}\right];$$

and

$$\underline{m}^{(\alpha)'} = [(n_\Delta^{(\alpha)} - 1) + (n_\Gamma^{(\alpha)} - 1) + (n_\Delta^{(\alpha)} + n_\Gamma^{(\alpha)} - 1)].$$

Then $D^{(\alpha)}$ becomes:

$$D^{(\alpha)'} = \frac{m^{(\alpha)'} - \underline{m}^{(\alpha)'}}{\overline{m}^{(\alpha)'} - \underline{m}^{(\alpha)'}}$$

and the generalized partition density is:

$$D' = \sum_{\alpha=1}^{c} \frac{1}{q^{(\alpha)}} \frac{n^{(\alpha)}}{N} D^{(\alpha)'}.$$

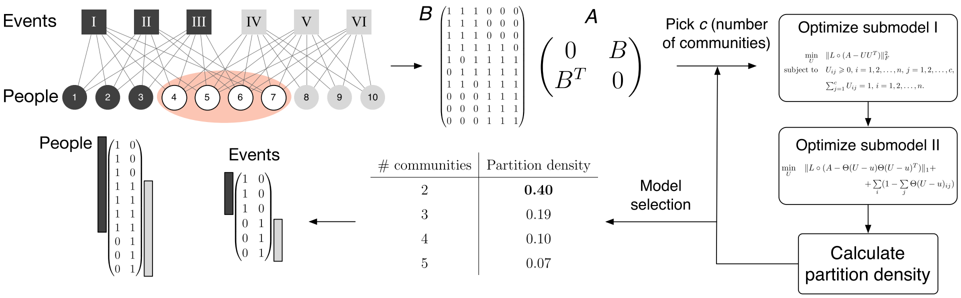
2.3. *An illustrative example*

We show a small example that illustrates how the method works. Figure 1 exhibits a bipartite network with two communities, which can be clearly recovered by our approach. Specifically, for $c = 2$ we have $m^{(1)} = 136$, $m^{(2)} = 114$; $\underline{m}^{(1)} = 35$, $\underline{m}^{(2)} = 35$; $\overline{m}^{(1)} = 147$, $\overline{m}^{(2)} = 147$; $q^{(1)} = 2$, $q^{(2)} = 2$ and $N = 20$. Let us illustrate how we can incorporate existing knowledge. If we know that Nodes III and IV are in the same community, then we can revise A and L such that the elements in the positions of $(13, 14)$ and $(14, 13)$ are 1. The result for events is changed to

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}^T,$$

which group III and IV together.

Note that the bipartite network can be projected onto the Event part or onto the People part. Two events are connected if they have at least one common neighbor in the People part, resulting in a complete network containing six nodes. The loss of information is obvious and the community structures vanish, which means that the problem of community detection in bipartite networks is not reducible to unipartite case.


Events
I
II
III
IV
V
VI
People
1
2
3
4
5
6
7
8
9
10
B
$\begin{pmatrix} 1&1&1&0&0&0\\ 1&1&1&0&0&0\\ 1&1&1&0&0&0\\ 1&1&1&1&1&0\\ 1&0&1&1&1&1\\ 0&1&1&1&1&1\\ 1&1&0&1&1&1\\ 0&0&0&1&1&1\\ 0&0&0&1&1&1\\ 0&0&0&1&1&1 \end{pmatrix}$
A
$\begin{pmatrix} 0 & B \\ B^T & 0 \end{pmatrix}$
Pick c (number of communities)
Optimize submodel I
$\min_U \quad \|L \circ (A - UU^T)\|_F^2$
subject to $\quad U_{ij} \geqslant 0,\ i = 1, 2, \ldots, n,\ j = 1, 2, \ldots, c,$
$\sum_{j=1}^{c} U_{ij} = 1,\ i = 1, 2, \ldots, n.$
Optimize submodel II
$\min_U \quad \|L \circ (A - \Theta(U - u)\Theta(U - u)^T)\|_1 +$
$+ \sum_i (1 - \sum_j \Theta(U - u)_{ij})$
Calculate partition density
Model selection
People
$\begin{pmatrix} 1&0\\ 1&0\\ 1&0\\ 1&1\\ 1&1\\ 1&1\\ 1&1\\ 0&1\\ 0&1\\ 0&1 \end{pmatrix}$
Events
$\begin{pmatrix} 1&0\\ 1&0\\ 1&0\\ 0&1\\ 0&1\\ 0&1 \end{pmatrix}$

# communities	Partition density
2	**0.40**
3	0.19
4	0.10
5	0.07

Fig. 1. (Color online) Illustration of wSBMF method. The network consists of events and people and exhibits two overlapping groups where some individuals (4)–(7) belong to both communities.

2.4. *Possible extensions*

The wSBMF model can be naturally extended to M-partite networks, whose adjacency matrix can be split into $M \times M$ blocks:

$$A = \begin{bmatrix} \mathbf{0}_{\Lambda_1,\Lambda_1} & B_{\Lambda_1,\Lambda_2} & \mathbf{0}_{\Lambda_1,\Lambda_3} & \cdots & \mathbf{0}_{\Lambda_1,\Lambda_M} \\ B^T_{\Lambda_1,\Lambda_2} & \mathbf{0}_{\Lambda_2,\Lambda_2} & B_{\Lambda_2,\Lambda_3} & \cdots & \mathbf{0}_{\Lambda_2,\Lambda_M} \\ \mathbf{0}_{\Lambda_3,\Lambda_1} & B^T_{\Lambda_2,\Lambda_3} & \mathbf{0}_{\Lambda_3,\Lambda_3} & \cdots & \mathbf{0}_{\Lambda_3,\Lambda_M} \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ \mathbf{0}_{\Lambda_{M-1},\Lambda_1} & \mathbf{0}_{\Lambda_{M-1},\Lambda_2} & \cdots & \mathbf{0}_{\Lambda_{M-1},\Lambda_{M-1}} & B_{\Lambda_{M-1},\Lambda_M} \\ \mathbf{0}_{\Lambda_M,\Lambda_1} & \mathbf{0}_{\Lambda_M,\Lambda_2} & \cdots & B^T_{\Lambda_{M-1},\Lambda_M} & \mathbf{0}_{\Lambda_M,\Lambda_M} \end{bmatrix},$$

where $\mathbf{0}_{\Lambda_i,\Lambda_j}$ is null matrix of size $n_{\Lambda_i} \times n_{\Lambda_j}$, and

$$B_{\Lambda_i,\Lambda_{i+1}ab} = \begin{cases} 1, & \text{if } a \sim b,\ a \in \Lambda_i,\ b \in \Lambda_{i+1} \\ 0, & \text{if } a \not\sim b,\ a \in \Lambda_i,\ b \in \Lambda_{i+1},\ i = 1, 2, \ldots, M-1. \end{cases}$$

In this case, L should be reformulated as:

$$L = \begin{bmatrix} \mathbf{0}_{\Lambda_1,\Lambda_1} & \mathbf{I}_{\Lambda_1,\Lambda_2} & \mathbf{0}_{\Lambda_1,\Lambda_3} & \cdots & \mathbf{0}_{\Lambda_1,\Lambda_M} \\ \mathbf{I}_{\Lambda_2,\Lambda_1} & \mathbf{0}_{\Lambda_2,\Lambda_2} & \mathbf{I}_{\Lambda_2,\Lambda_3} & \cdots & \mathbf{0}_{\Lambda_2,\Lambda_M} \\ \mathbf{0}_{\Lambda_3,\Lambda_1} & \mathbf{I}_{\Lambda_3,\Lambda_2} & \mathbf{0}_{\Lambda_3,\Lambda_3} & \cdots & \mathbf{0}_{\Lambda_3,\Lambda_M} \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ \mathbf{0}_{\Lambda_{M-1},\Lambda_1} & \mathbf{0}_{\Lambda_{M-1},\Lambda_2} & \cdots & \mathbf{0}_{\Lambda_{M-1},\Lambda_{M-1}} & \mathbf{I}_{\Lambda_{M-1},\Lambda_M} \\ \mathbf{0}_{\Lambda_M,\Lambda_1} & \mathbf{0}_{\Lambda_M,\Lambda_2} & \cdots & \mathbf{I}_{\Lambda_M,\Lambda_{M-1}} & \mathbf{0}_{\Lambda_M,\Lambda_M} \end{bmatrix},$$

where $\mathbf{I}_{\Lambda_i,\Lambda_j}$ is matrix where all entries are one with size $n_{\Lambda_i} \times n_{\Lambda_j}$.

3. Results

In this section, we evaluate the performance of our method using both synthetic and real-world networks.

3.1. *Datasets description*

We first discuss the existing bipartite benchmark networks.[11] The benchmark has five communities, each having the same number of nodes. Edges only exist between Δ and Γ with possibility p_{in} if they are in the same community and p_{out} if otherwise. Often, p_{in} is set equal to either 0.5 or 0.9 and p_{out} is set as αp_{in}, where α varies from 0 to 1. With increasing α, the community structure becomes less clear. Here, we propose two new, more realistic benchmark graphs that exhibit overlaps, variable community sizes, and fixed density with different mixing parameters.

- Nonoverlapping communities: This class of networks has four communities with the same number of nodes (each with 32 from Δ and 32 from Γ). Edges exist only between Δ and Γ. On average, each node has $Z_{\text{in}} + Z_{\text{out}} = 16$ edges. In other words, each node in Δ has Z_{in} neighbors within its own community and Z_{out} ones outside. With decreasing Z_{out}, the community structures become clearer.
- Overlapping communities: This class of networks has c communities and the number of nodes in each community can differ from each other. A community α contains $n_{\Delta}^{(\alpha)}$ nodes and $n_{\Gamma}^{(\alpha)}$ ones in Δ and Γ, respectively. On average each Δ node in the community α has $Z_{\text{in}}^{(\alpha)}\Gamma$ neighbors in its own community and $Z_{\text{out}}^{(\alpha)}\Gamma$ neighbors in other communities. Actually, since we should have $Z_{\text{in}}^{(\alpha)}/n_{\Gamma}^{(\alpha)} = Z_{\text{in}}^{(\alpha')}/n_{\Gamma}^{(\alpha')}$, and $Z_{\text{out}}^{(\alpha)}/(\sum_t n_{\Gamma}^{(t)} - n_{\Gamma}^{(\alpha)}) = Z_{\text{in}}^{(\alpha\prime)}/(\sum_t n_{\Gamma}^{(t)} - n_{\Gamma}^{(\alpha\prime)})$, $\alpha, \alpha' = 1, 2, \ldots c$, it is enough only to give $Z_{\text{in}}^{(1)}$ and $Z_{\text{out}}^{(1)}$ to generate the network. In our setting there are four communities containing 32Δ nodes and 32Γ ones in each community. In addition, there are t overlapping Δ nodes between communities α and $\alpha + 1$, $\alpha = 1, 2, 3$. $Z_{\text{in}}^{(1)}$ and $Z_{\text{out}}^{(1)}$ are set to 10 and 6, respectively.

We also use real-world networks for evaluation.

- Southern women network[29]: This dataset is the network describing the relations between 18 women and 14 social events. Edges only exist between the women and the events, which makes the graph bipartite. There are 89 edges. The network is commonly used as a benchmark for bipartite community detection.
- Senator network[a]: This is the network of 110 US senators connected by voting records for 696 bills. There is an edge between the senator and the bill if the senator voted for the bill. We remove inactive senators who abstained from more than 30% of the bills and also the inactive bills which are waived by more than 30% of senators. The final dataset contains 96 senators and 690 bills. There are still abstention cases in the network, which are considered as missing values and can be handled by L.

3.2. *Assessment standards*

Normalized mutual information is used as the standard to evaluate community structure detection performance. The value can be formulated as follows[30]:

$$I_{\text{norm}}(M_1, M_2) = \frac{\sum_{i=1}^{c}\sum_{j=1}^{c} n_{ij} \ln \frac{n_{ij} n}{n_i^{(1)} n_j^{(2)}}}{\sqrt{\left(\sum_{i=1}^{c} n_i^{(1)} \ln \frac{n_i^{(1)}}{n}\right)\left(\sum_{j=1}^{c} n_j^{(2)} \ln \frac{n_j^{(2)}}{n}\right)}},$$

[a] http://www.senate.gov/.

where M_1 and M_2 are the true cluster label and the computed cluster label, respectively; c is the community number; n is the number of nodes; n_{ij} is the number of nodes in the true cluster i that are assigned to the computed cluster j; $n_i^{(1)}$ is the number of nodes in the true cluster i and $n_j^{(2)}$ is the number of nodes in the computed cluster j. The larger the values of NMI, the better the graph partitioning results. For overlapping benchmarks we use the generalized normalized mutual information.[31]

3.3. *Results*

We compare our method with the BRIM model,[11] which is the only method that we can get the codes, on the synthetic benchmarks. Note the BRIM method cannot handle overlapping communities and missing values in the network. To show that the problem of detecting overlapping communities in bipartite networks is not trivial and cannot be reduced to the unipartite case, we also compare our method with SBMF model[22] on the two unipartite networks Δ and Γ, where the two nodes are connected if they have at least one common neighbor.

In many real scenarios there is background information available. We can incorporate it into the detection process by revising the objective matrix A and the weight matrix L to improve the performance of detection and the interpretability of the results. Specifically, we consider two types of background information for node pairs of the same type (i.e. Δ or Γ): (i) `existence constraint` C_e: $(i,j) \in C_e$ means that nodes i and j are connected; (ii) `absence constraint` C_a: $(i,j) \in C_a$ means that nodes i and j are not connected.

We only consider incorporating background information on the nodes in Δ in this paper for simplicity. Given a bipartite network with n_Δ nodes in Δ, there are $n_\Delta(n_\Delta - 1)/2$ pairs of nodes available. We randomly select 5% of pairs for prior information: if the two nodes in one pair have the same community label, we assume that they belong to C_e, otherwise they belong to C_a.[32,33] The zero matrices $\mathbf{0}_\Gamma$ in A and L are revised accordingly:

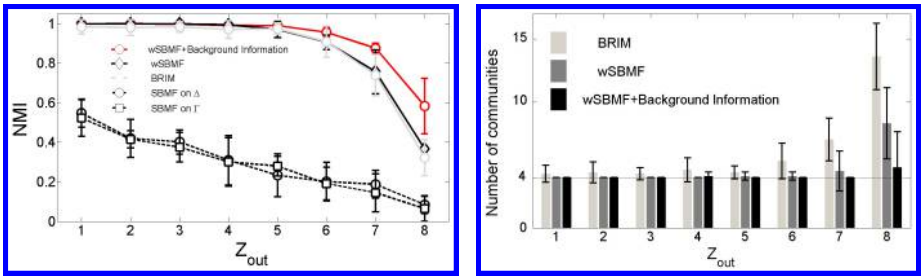
$$\mathbf{0}_{\Delta ij} = \begin{cases} 1, & \text{if } (i,\, j) \in C_e, \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

where $\mathbf{0}_\Delta$ is the submatrix in A.

$$\mathbf{0}_{\Delta\, ij} = \begin{cases} \gamma, & \text{if } (i,\, j) \in C_e \text{ or } (i,\, j) \in C_a, \\ 0, & \text{otherwise,} \end{cases} \tag{7}$$

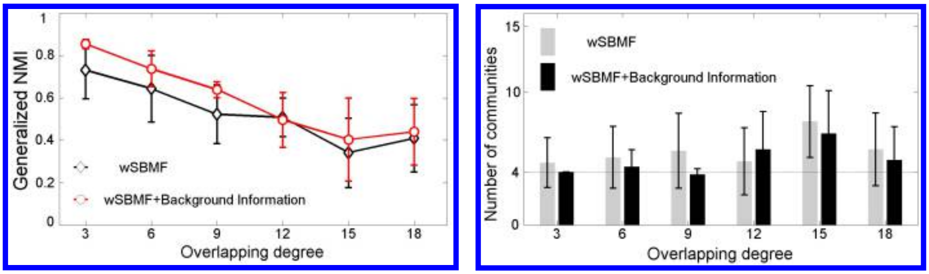
where $\mathbf{0}_\Delta$ is the submatrix in L. We set γ equal to 1.

The results are shown in Figs. 2 and 3. They show that the wSBMF method is much better than SBMF on unipartite networks, indicating the nonreducible property of community detection problem in bipartite networks, and it also performs better than BRIM in nonoverlapping community benchmark graphs. Our



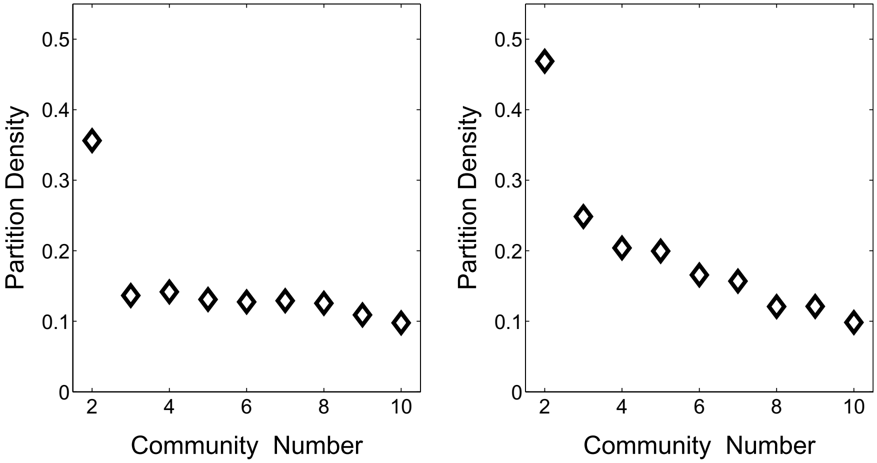
(a) (b)

Fig. 2. (Color online) Performance of BRIM and wSBMF on the bipartite networks, SBMF on the monopartite networks, and the number of communities estimated by BRIM and wSBMF on nonoverlapping networks. We randomly select 5% of pairs in Δ for background information.



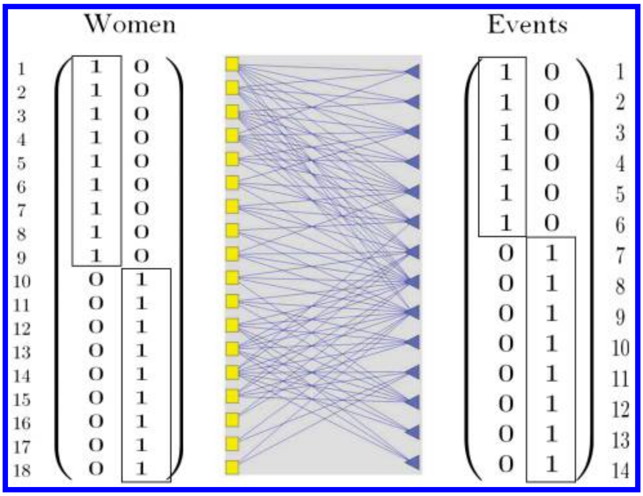
(a) (b)

Fig. 3. (Color online) Performance of wSBMF and the number of communities estimated by SBMF on overlapping networks. We randomly select 5% of pairs in Δ for background information.



(a) (b)

Fig. 4. Averaged partition density of wSBMF versus community number on (a) women network and (b) senator network.



Women		
1	1	0
2	1	0
3	1	0
4	1	0
5	1	0
6	1	0
7	1	0
8	1	0
9	1	0
10	0	1
11	0	1
12	0	1
13	0	1
14	0	1
15	0	1
16	0	1
17	0	1
18	0	1

Events		
1	0	1
1	0	2
1	0	3
1	0	4
1	0	5
1	0	6
0	1	7
0	1	8
0	1	9
0	1	10
0	1	11
0	1	12
0	1	13
0	1	14

Fig. 5. (Color online) Communities detected by wSBMF model in the women network. There are no outliers and overlapping nodes.

method can identify reasonable number of communities, and the background information can significantly improve the results. We also evaluate the method on the southern women network and the senator network. Figure 4 shows the results of partition density under different community numbers on the two networks, and the most appropriate number is 2 for both of them. For the southern women network, the result is very similar to that in Ref. 29, where there are two groups in women, women $1-9$ and $9-18$. For the senator network, the result is consistent with American two-party politics. Figure 5 shows the result of community structure on the women network detected by wSBMF. We also use exponential entropy e^{H_i}, $i = 1, 2, \ldots, n_{\Delta}$,[34] to analyze the strength of women's community

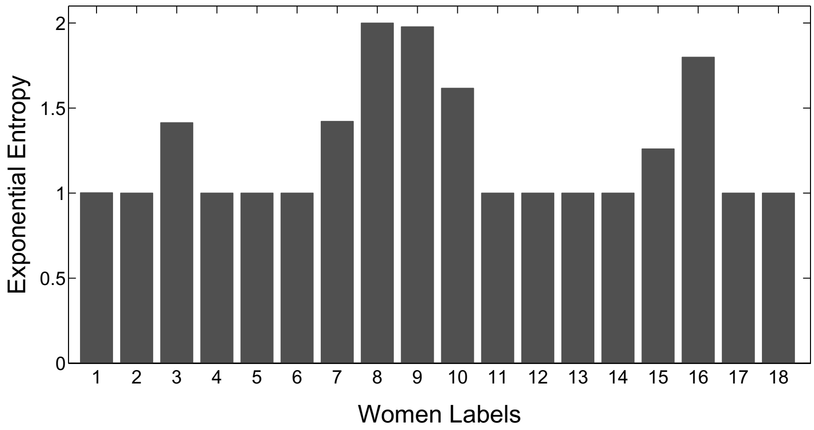


Fig. 6. Exponential entropy of women. Higher value means fuzzier membership degree.

memberships, where

$$H_i = -\sum_{j=1}^{2} U_{ij} \log U_{ij}, \quad i = 1, 2, \ldots, n_\Delta.$$

The result is given in Fig. 6.

4. Discussion

In this paper, we have shown how to apply SBMF and partition density to find communities in bipartite networks. The model is parameter free, easy to implement, and flexible enough to incorporate background information. Experimental results on both the synthetic and real-world networks demonstrate the effectiveness of the proposed method.

There are two interesting problems for future work: (i) extension of the method to weighted bipartite networks and directed bipartite networks; and (ii) theoretical investigation on partition density and algorithm design for its direct optimization.

Acknowledgment

Z.-Y. Zhang was supported by the National Natural Science Foundation of China under Grant No. 61203295, 11401602 and 71401189.

Appendix

Summarization of Algorithms 1 and 2. We set the iteration number C_1 equal to 10 and the iteration number C_2 equal to 100.

Algorithm 1 Nonnegative Matrix Factorization (Alternative Least Squares Error)

```
Require: $B, C_1$
Ensure: $U_1, U_2$
Initialize elements of $U_1$ with nonnegative random numbers drawn from $[0, 1]$.
for $t = 1 : C_1$ do
    Solve for $U_2$ in equation $U_1^T U_1 U_2 = U_1^T A$
    $U_2 = \max(U_2, 0)$
    Solve for $U_2$ in equation $U_2 U_2^T U_1^T = U_2 A^T$
    $U_1 = \max(U_1, 0)$
end for
```

Algorithm 2 Weighted Symmetric Nonnegative Matrix Factorization (Multiplicative Updates)

```
Require: $A, U, \mathrm{C}_2$
Ensure: $U$
for $t = 1 : \mathrm{C}_2$ do
    $U := U \circ \frac{[(L \circ A)U]}{[L \circ (UU^T)U]}$
end for
$U_{ij} := \frac{U_{ij}}{\sum_j U_{ij}}, \; i = 1, 2, \ldots, n$
```

References

1. M. Girvan and M. E. J. Newman, *Proc. Natl. Acad. Sci.* **99**, 7821 (2002).
2. M. E. J. Newman and M. Girvan, *Physical Rev. E* **69**, 026113 (2004).
3. M. Dreze *et al.*, *Science* **333**, 601 (2011).
4. M. E. J. Newman, *Proc. Natl. Acad. Sci.* **103**, 8577 (2006).
5. Y.-Y. Ahn, J. P. Bagrow and S. Lehmann, *Nature* **466**, 761 (2010).
6. N. Gulbahce and S. Lehmann, *BioEssays* **30**, 934 (2008).
7. M. A. Porter, J.-P. Onnela and P. J. Mucha, *Notices Amer. Math. Soc.* **56**, 1082 (2009).
8. S. Fortunato, *Phys. Rep.* **486**, 75 (2010).
9. X. Wu and Z. Liu, *Physica A* **387**, 623 (2008).
10. L. Weng, F. Menczer and Y.-Y. Ahn, arXiv:1306.0158.
11. M. J. Barber, *Phys. Rev. E* **76**, 066102 (2007).
12. N. Du, B. Wang, B. Wu and Y. Wang, *IEEE/WIC/ACM Int. Conf. on Web Intelligence and Intelligent Agent Technology*, 2008 *(WI-IAT'08)*, Vol. 1 (IEEE, 2008), pp. 176–179.
13. W. Zhan, Z. Zhang, J. Guan and S. Zhou, *Phys. Rev. E* **83**, 066120 (2011).
14. S. Lehmann, M. Schwartz and L. K. Hansen, *Phys. Rev. E* **78**, 016108 (2008).
15. X. Liu and T. Murata, *IEEE/WIC/ACM Int. Joint Conf. on Web Intelligence and Intelligent Agent Technologies*, 2009 (WI-IAT'09), Vol. 1 (IET, 2009), pp. 50–57.
16. P. G. Lind, M. C. González and H. J. Herrmann, *Phys. Rev. E* **72**, 056127 (2005).
17. P. G. Lind and H. J. Herrmann, *New J. Phys.* **9**, 228 (2007).
18. Y. Y. Ahn, S. E. Ahnert, J. P. Bagrow and A.-L. Barabási, *Sci. Rep.* **1** (2011).
19. H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai and A.-L. Barabási, *Nature* **407**, 651 (2000).
20. M. E. J. Newman, *Phys. Rev. E* **64**, 016132 (2001).
21. T. Zhou, J. Ren, M. Medo and Y.-C. Zhang, *Phys. Rev. E* **76**, 046115 (2007).
22. Z. Y. Zhang, Y. Wang and Y.-Y. Ahn, *Phys. Rev. E* **87**, 062803 (2013).
23. H. Lee, J. Yoo and S. Choi, *IEEE Signal Process. Lett.* **17**, 4 (2010).
24. P. Paatero and U. Tapper, *Environmetrics* **5**, 111 (1994).
25. M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca and R. J. Plemmons, *Comput. Statist. Data Anal.* **52**, 155 (2007).
26. Z.-Y. Zhang, C. Ding, T. Li and X. Zhang, *Seventh IEEE Int. Conf. on Data Mining,* 2007 (ICDM 2007) (IEEE, 2007), pp. 391–400.
27. Z.-Y. Zhang, T. Li, C. Ding, X. Ren and X. Zhang, *Data Min. Knowl. Discov.* **20**, 28 (2010).
28. Z.-Y. Zhang, *Data Mining: Foundations and Intelligent Paradigms* (Springer, 2012), pp. 99–134.

29. A. Davis, B. B. Gardner and M. R. Gardner, *Deep South* (University of Chicago Press, Chicago, 1941).
30. A. Strehl and J. Ghosh, *J. Mach. Learn. Res.* **3**, 583 (2002).
31. A. Lancichinetti, S. Fortunato and J. Kertész, *New J. Phys.* **11**, 033015 (2009).
32. Z.-Y. Zhang, *Europhys. Lett.*, 48005+ (2013).
33. P. K. Gopalan and D. M. Blei, *Proc. Natl. Acad. Sci.* **110**, 14534 (2013).
34. L. Campbell, *Probab. Theory Related Fields* **5**, 217 (1966).