A Tensor Spectral Approach to Learning Mixed Membership Community Models

Animashree Anandkumar A.ANANDKUMAR@UCI.EDU
University of California, Irvine

Rong Ge RONGGE@CS.PRINCETON.EDU
Princeton University

Daniel Hsu DAHSU@MICROSOFT.COM
Microsoft Research

Sham M. Kakade SKAKADE@MICROSOFT.COM
Microsoft Research

Abstract

Detecting hidden communities from observed interactions is a classical problem. Theoretical analysis of community detection has so far been mostly limited to models with non-overlapping communities such as the stochastic block model. In this paper, we provide guaranteed community detection for a family of probabilistic network models with overlapping communities, termed as the mixed membership Dirichlet model, first introduced in Airoldi et al. (2008). This model allows for nodes to have fractional memberships in multiple communities and assumes that the community memberships are drawn from a Dirichlet distribution. Moreover, it contains the stochastic block model as a special case. We propose a unified approach to learning communities in these models via a tensor spectral decomposition approach. Our estimator uses low-order moment tensor of the observed network, consisting of 3-star counts. Our learning method is based on simple linear algebraic operations such as singular value decomposition and tensor power iterations. We provide guaranteed recovery of community memberships and model parameters, and present a careful finite sample analysis of our learning method. Additionally, our results match the best known scaling requirements for the special case of the (homogeneous) stochastic block model.

Keywords: Community detection, spectral methods, tensor methods, moment-based estimation, mixed membership models.

1. Introduction[1]

Studying communities forms an integral part of social network analysis. A community generally refers to a group of individuals with shared interests (e.g. music, sports), or relationships (e.g. friends, co-workers). Various probabilistic and non-probabilistic network models attempt to explain community formation. In addition, they also attempt to quantify interactions and the extent of overlap between different communities, relative sizes among

[1] Part of this work was done when AA and RG were visiting MSR New England. AA is supported in part by the NSF Career award CCF-1254106, NSF Award CCF-1219234, AFOSR Award FA9550-10-1-0310 and the ARO Award W911NF-12-1-0404.

the communities, and various other network properties. Studying such community models is also of interest in other domains, e.g. in biological networks.

While there exists a vast literature on community models, learning these models is typically challenging, and various heuristics such as Markov Chain Monte Carlo (MCMC) or variational expectation maximization (EM) are employed in practice. Such heuristics tend to be unreliable and scale poorly for large networks. On the other hand, community models with guaranteed learning methods tend to be restrictive. A popular class of probabilistic models, termed as the *stochastic blockmodels*, have been widely studied and enjoy strong theoretical learning guarantees, e.g. (White et al., 1976; Holland et al., 1983; Fienberg et al., 1985; Wang and Wong, 1987; Snijders and Nowicki, 1997; McSherry, 2001). However, they posit that an individual belongs to a single community, which does not hold in most real settings (Palla et al., 2005).

In this paper, we consider a class of mixed membership community models, originally introduced by Airoldi et al. (2008), and recently employed by Xing et al. (2010) and Gopalan et al. (2012). This model has been shown to be effective in many real-world settings, but so far, no learning approach exists with provable guarantees. In this paper, we provide a novel learning approach for learning these models and establish regimes where the communities can be recovered efficiently. The mixed membership community model of Airoldi et al. (2008) has a number of attractive properties. It retains many of the convenient properties of the stochastic block model. For instance, conditional independence of the edges is assumed, given the community memberships of the nodes in the network. At the same time, it allows for communities to overlap, and for every individual to be fractionally involved in different communities. It includes the stochastic block model as a special case (corresponding to zero overlap among the different communities). This enables us to compare our learning guarantees with existing works for stochastic block models, and also study how the extent of overlap among different communities affects the learning performance.

1.1. Summary of Results

We now summarize the main contributions of this paper. We propose a novel approach for learning mixed membership community models of Airoldi et al. (2008). Our approach is a method-of-moments estimator and incorporates tensor spectral decomposition techniques. We provide guarantees for our approach under a set of sufficient conditions. Finally, we compare our results to existing ones for the special case of the stochastic block model, where nodes belong to a single community.

Learning general mixed membership models: We present a unified approach for the mixed membership model of Airoldi et al. (2008). The extent of overlap between different communities in this model class is controlled (roughly) through a single scalar parameter, termed as the Dirichlet concentration parameter $\alpha_0 := \sum_i \alpha_i$, when the community membership vectors are drawn from the Dirichlet distribution $\mathrm{Dir}(\alpha)$. When $\alpha_0 \to 0$, the mixed membership model degenerates to a stochastic block model. We propose a unified learning method for the class of mixed membership models. We provide explicit scaling requirements in terms of the extent of community overlaps (through α_0), the network size n, the number of communities k, and the average edge connectivity across various communities. For instance, for the special case, where p is the probability of an intra-community edge, and q corresponds to the probability of inter-community connectivity, when the average community sizes are equal, we require that[2]

$$n = \tilde{\Omega}(k^2(\alpha_0+1)^2), \qquad \frac{p-q}{\sqrt{p}} = \tilde{\Omega}\left(\frac{(\alpha_0+1)k}{n^{1/2}}\right). \tag{1}$$

Thus, we require n to be large enough compared to the number of communities k, and for the separation $p-q$ to be large enough, so that the learning method can distinguish the different communities. Moreover, we see that the scaling requirements become more stringent as α_0 increases. This is intuitive since it is harder to learn communities with more overlap, and we quantify this scaling. We also quantify the error bounds for estimating various parameters of the mixed membership model. Lastly, we establish zero-error guarantees for support recovery: our learning method correctly identifies (w.h.p) all the significant memberships of a node and also identifies the set of communities where a node does not have a strong presence.

Learning Stochastic Block Models and Comparison with Previous Results: For the special case of stochastic block models ($\alpha_0 \to 0$), the scaling requirements in (2) reduces to

$$n = \tilde{\Omega}(k^2), \qquad \frac{p-q}{\sqrt{p}} = \tilde{\Omega}\left(\frac{k}{n^{1/2}}\right), \tag{2}$$

The above requirements match the best known bounds[3] (up to poly-log factors), and were previously achieved by Yudong et al. (2012) via convex optimization. In contrast, we propose an iterative non-convex approach involving tensor power iterations and linear algebraic techniques, and obtain similar guarantees for the stochastic block model. For a detailed comparison of learning guarantees under various methods for learning stochastic block models, see (Yudong et al., 2012).

Thus, we provide guaranteed recovery of the communities under the mixed membership model, and our scaling requirements in (1) explicitly incorporate the extent of community overlaps. Many real-world networks involve sparse community memberships and the total number of communities is typically much larger than the extent of membership of a single individual, e.g. hobbies/interests of a person, university/company networks that a person belongs to, the set of transcription factors regulating a gene, and so on. Thus, we see that in this regime of practical interest, where $\alpha_0 = \Theta(1)$, the scaling requirements in (1) match those of the stochastic block model in (2) (up to polylog factors) without any degradation in learning performance. Thus, we establish that learning community models with sparse community memberships is akin to learning stochastic block models, and we present a unified learning approach and analysis for these models. To the best of our knowledge, this work is the first to establish polynomial time learning guarantees for probabilistic network models with overlapping communities, and we provide a fast and an iterative learning approach through linear algebraic techniques and tensor power iterations.

[2] The notation $\tilde{\Omega}(\cdot), \tilde{O}(\cdot)$ denotes $\Omega(\cdot), O(\cdot)$ up to poly-log factors.

[3] There are many methods which achieve the best known scaling for n in (2), but have worse scaling for the separation $p-q$. This includes variants of the spectral clustering method, e.g. (Chaudhuri et al., 2012). See (Yudong et al., 2012) for a detailed comparison.

1.2. Overview of Techniques

We now describe the main techniques employed in our learning approach and in establishing the recovery guarantees.

Method of moments and subgraph counts: We propose an efficient learning algorithm based on low order moments, viz., counts of small subgraphs. Specifically, we employ a third-order tensor which counts the number of 3-stars in the observed network. A 3-star is a star graph with three leaves and we count the occurrences of such 3-stars across different groups of nodes. We establish that (suitably adjusted) 3-star count tensor has a simple relationship with the model parameters, when the network is drawn from a mixed membership community model. In particular, we propose a multi-linear transformation (termed as whitening) under which the *canonical polyadic (CP) decomposition* of the tensor yields the model parameters and the community vectors. Note that the decomposition of a general tensor into its rank-one components is referred to as its CP decomposition (Kolda and Bader, 2009) and is in general NP-hard (Hillar and Lim, 2012). However, we reduce our learning problem to an orthogonal symmetric tensor decomposition, for which tractable decomposition exists, as described below.

Tensor spectral decomposition via power iterations: Our tensor decomposition method is based on the popular tensor power iterations, e.g. see (Anandkumar et al., 2012a). It is a simple iterative method to compute the stable eigen-pairs of a tensor. In this paper, we propose various modifications to the basic power method to strengthen the recovery guarantees under perturbations. For instance, we introduce a novel adaptive deflation techniques. We optimize performance for the regime where the community overlaps are small.

Sample analysis: We establish that our learning approach correctly recovers the model parameters and the community memberships of all nodes under exact moments. We then carry out a careful analysis of the empirical graph moments, computed using the network observations. We establish tensor concentration bounds and also control the perturbation of the various quantities used by our learning algorithm via matrix Bernstein's inequality (Tropp, 2012, thm. 1.4) and other inequalities. We impose the scaling requirements in (1) for various concentration bounds to hold.

1.3. Related Work

Many algorithms provide learning guarantees for stochastic block models. A popular method is based on spectral clustering (McSherry, 2001), where community memberships are inferred through projection onto the spectrum of the Laplacian matrix (or its variants). This method is fast and easy to implement (via singular value decomposition). There are many variants of this method, e.g. the work by Chaudhuri et al. (2012) employs normalized Laplacian matrix to handle degree heterogeneities. In contrast, the work of (Yudong et al., 2012) uses convex optimization techniques via semi-definite programming learning block models. For a detailed comparison of learning guarantees under various methods for learning stochastic block models, see Yudong et al. (2012). Recently, some non-probabilistic approaches have been introduced with overlapping community models by Arora et al. (2012) and Balcan et al. (2012). However, their setting is considerably different than the one in this paper. We leverage the recent developments from Anandkumar et al. (2012c,a,b) for learning topic models and other latent variable models based on the method of moments. They consider learning these models from second- and third-order observed moments through linear algebraic and tensor-based techniques. We exploit the tensor power iteration method of Anandkumar et al. (2012b) and provide additional improvements to obtain stronger recovery guarantees. Moreover, the sample analysis is quite different in the community setting compared to other latent variable models analyzed in the previous works.

2. Community Models and Graph Moments

2.1. Community Membership Models

Notation: We consider network with n nodes and let $[n] := \{1, 2, \dots, n\}$. Let G be the $\{0, 1\}$ adjacency[4] matrix for the random network and let $G_{A,B}$ be the submatrix of G corresponding to rows $A \subseteq [n]$ and columns $B \subseteq [n]$. For node i, let $\pi_i \in \mathbb{R}^k$ denote its *community membership vector*. Define $\Pi := [\pi_1|\pi_2|\cdots|\pi_n] \in \mathbb{R}^{k\times n}$. and let $\Pi_A := [\pi_i : i \in A] \in \mathbb{R}^{k\times|A|}$ denote the set of column vectors restricted to $A \subseteq [n]$. For a matrix M, let $(M)_i$ and $(M)^i$ denote its i^{th} column and row respectively. For a matrix M with singular value decomposition (SVD) $M = UDV^\top$, let $(M)_{k-svd} := U\tilde{D}V^\top$ denote the k-rank SVD of M, where $\tilde{D}$ is limited to top-k singular values of M. Let $M^\dagger$ denote the MoorePenrose pseudo-inverse of M. Let $\mathbb{I}(\cdot)$ be the indicator function. We use the term high probability to mean with probability $1 - n^{-c}$ for any constant $c > 0$.

Mixed membership model: In this model, the community membership vector π_u at node u is a probability vector, i.e., $\sum_{i\in[k]} \pi_u(i) = 1$, for all $u \in [n]$. Given the community membership vectors, the generation of the edges is as follows: given vectors π_u and π_v, the probability of an edge from[5] u to v is $\pi_u^\top P\pi_v$, and the edges are independently drawn. Here, $P \in [0, 1]^{k\times k}$ and we refer to it as the *community connectivity matrix*. We consider the setting of Airoldi et al. (2008), where the community vectors $\{\pi_u\}$ are i.i.d. draws from the Dirichlet distribution, denoted by $\mathrm{Dir}(\alpha)$, with parameter vector $\alpha \in \mathbb{R}^k_{>0}$. The probability density function is given by

$$\mathbb{P}[\pi] = \frac{\prod_{i=1}^k \Gamma(\alpha_i)}{\Gamma(\alpha_0)} \prod_{i=1}^k \pi_i^{\alpha_i - 1}, \quad \pi \sim \mathrm{Dir}(\alpha), \alpha_0 := \sum_i \alpha_i, \tag{3}$$

where $\Gamma(\cdot)$ is the Gamma function and the ratio of the Gamma function serves as the normalization constant.

Let $\widehat{\alpha}$ denote the normalized parameter vector α/α_0, where $\alpha_0 := \sum_i \alpha_i$. In particular, note that $\widehat{\alpha}$ is a probability vector: $\sum_i \widehat{\alpha}_i = 1$. Intuitively, $\widehat{\alpha}$ denotes the relative expected sizes of the communities (since $\mathbb{E}[n^{-1} \sum_{u\in[n]} \pi_u[i]] = \widehat{\alpha}_i$). Let $\widehat{\alpha}_{\max}$ be the largest entry in $\widehat{\alpha}$, and $\widehat{\alpha}_{\min}$ be the smallest entry. Our learning guarantees will depend on these parameters.

The stochastic block model is a limiting case of the mixed membership model when the Dirichlet parameter is $\alpha = \alpha_0 \cdot \widehat{\alpha}$, where the probability vector $\widehat{\alpha}$ is held fixed and $\alpha_0 \to 0$.

[4] Our analysis can easily be extended to weighted adjacency matrices with bounded entries.

[5] We consider directed networks in this paper, but note that the results also hold for undirected community models, where P is a symmetric matrix, and an edge (u, v) is formed with probability $\pi_u^\top P\pi_v = \pi_v^\top P\pi_u$.

In this case, the community membership vectors π_i correspond to coordinate basis vectors. In the other extreme when $\alpha_0 \to \infty$, the Dirichlet distribution becomes peaked around a single point, for instance, if $\alpha_i \equiv c$ and $c \to \infty$, the Dirichlet distribution is peaked at $k^{-1} \cdot \mathbf{1}$, where $\mathbf{1}$ is the all-ones vector. Thus, the parameter α_0 controls the extent of overlap among different communities.

2.2. Graph Moments Under Mixed Membership Models

Our approach for learning a mixed membership community model relies on the form of the graph moments[6] under the mixed membership model. We now describe the specific graph moments used by our learning algorithm (based on 3-star and edge counts) and provide explicit forms for the moments, assuming draws from a mixed membership community model.

Notations: Recall that G denotes the adjacency matrix, and that $G_{X,A}$ denotes the submatrix corresponding to edges going from X to A. Recall that $P \in [0,1]^{k\times k}$ denotes the community connectivity matrix. Define

$$F := \Pi^\top P^\top = [\pi_1|\pi_2|\cdots|\pi_n]^\top P^\top. \tag{4}$$

For a subset $A \subseteq [n]$ of individuals, let $F_A \in \mathbb{R}^{|A|\times k}$ denote the submatrix of F corresponding to nodes in A, *i.e.*, $F_A := \Pi_A^\top P^\top$. Let $\operatorname{Diag}(v)$ denote a diagonal matrix with diagonal entries given by a vector v. Our learning algorithm uses moments up to the third-order, represented as a tensor. A third-order tensor T is a three-dimensional array whose (p,q,r)-th entry denoted by $T_{p,q,r}$. The symbol $\otimes$ denotes the standard Kronecker product: if u, v, w are three vectors, then

$$(u \otimes v \otimes w)_{p,q,r} := u_p \cdot v_q \cdot w_r. \tag{5}$$

3-star counts: The primary quantity of interest is a third-order tensor which counts the number of 3-stars. A 3-star is a star graph with three leaves $\{a,b,c\}$ and we refer to the internal node x of the star as its "head", and denote the structure by $x \to \{a,b,c\}$. We partition the network into four parts and consider 3-stars such that each node in the 3-star belongs to a different partition. Consider a partition[7] A,B,C,X of the network. We count the number of 3-stars from X to A,B,C, and our quantity of interest is

$$\mathrm{T}_{X\to\{A,B,C\}} := \frac{1}{|X|}\sum_{i\in X}[G_{i,A}^\top \otimes G_{i,B}^\top \otimes G_{i,C}^\top], \tag{6}$$

where $\otimes$ is the Kronecker product, defined in (5), and $G_{i,A}$ is the row vector supported on the set of neighbors of i belonging to set A. Define

$$\mu_{X\to A} := \frac{1}{|X|}\sum_{i\in X}[G_{i,A}^\top], \quad G_{X,A}^{\alpha_0} := \left(\sqrt{\alpha_0+1}G_{X,A} - (\sqrt{\alpha_0+1}-1)\mathbf{1}\mu_{X\to A}^\top\right). \tag{7}$$

[6] We interchangeably use the term first order moments for edge counts and third order moments for 3-star counts.

[7] For our theoretical guarantees to hold, the partitions A,B,C,X can be randomly chosen and are of size $\Theta(n)$.

Similarly, we define[8] adjusted third-order statistics, $\mathrm{T}^{\alpha_0}_{X\to\{A,B,C\}}$ given by

$$\begin{aligned}&(\alpha_0+1)(\alpha_0+2)\,\mathrm{T}_{X\to\{A,B,C\}}+2\,\alpha_0^2\,\mu_{X\to A}\otimes\mu_{X\to B}\otimes\mu_{X\to C}\\&-\frac{\alpha_0(\alpha_0+1)}{|X|}\sum_{i\in X}\left[G_{i,A}^\top\otimes G_{i,B}^\top\otimes\mu_{X\to C}+G_{i,A}^\top\otimes\mu_{X\to B}\otimes G_{i,C}^\top+\mu_{X\to A}\otimes G_{i,B}^\top\otimes G_{i,C}^\top\right],\end{aligned}\tag{8}$$

and it reduces to the (scaled version of) 3-star count $\mathrm{T}_{X\to\{A,B,C\}}$ defined in (6) for the stochastic block model ($\alpha_0\to 0$).

Proposition 1 (Moments in Mixed Membership Model) *Given partitions A,B,C,X and $G^{\alpha_0}_{X,A}$ and T^{α_0}, as in (7) and (8), normalized Dirichlet concentration vector $\widehat{\alpha}$, and $F:=\Pi^\top P^\top$, where P is the community connectivity matrix and Π is the matrix of community memberships, we have*

$$\mathbb{E}[(G^{\alpha_0}_{X,A})^\top|\Pi_A,\Pi_X]=F_A\,\mathrm{Diag}(\widehat{\alpha}^{1/2})\Psi_X,\tag{9}$$

$$\mathbb{E}[\mathrm{T}^{\alpha_0}_{X\to\{A,B,C\}}|\Pi_A,\Pi_B,\Pi_C]=\sum_{i=1}^{k}\widehat{\alpha}_i(F_A)_i\otimes(F_B)_i\otimes(F_C)_i,\tag{10}$$

where $(F_A)_i$ corresponds to i^{th} column of F_A and Ψ_X relates to the community membership matrix Π_X as

$$\Psi_X:=\mathrm{Diag}(\widehat{\alpha}^{-1/2})\left(\sqrt{\alpha_0+1}\Pi_X-(\sqrt{\alpha_0+1}-1)\left(\frac{1}{|X|}\sum_{i\in X}\pi_i\right)\mathbf{1}^\top\right).$$

Moreover, we have that

$$|X|^{-1}\mathbb{E}_{\Pi_X}[\Psi_X\Psi_X^\top]=I.\tag{11}$$

3. Algorithm for Learning Mixed Membership Models

The simple form of the graph moments derived in the previous section is now utilized to recover the community vectors Π and model parameters $P,\widehat{\alpha}$ of the mixed membership model. The method is based on the so-called tensor power method, used to obtain a tensor decomposition. For a detailed discussion on the tensor power method, see (Anandkumar et al., 2012b). Below, we discuss the various steps of our algorithm.

Partitioning: We first partition the data into 5 disjoint sets A,B,C,X,Y. The set X is employed to compute whitening matrices $\hat{W}_A$, $\hat{W}_B$ and $\hat{W}_C$, described in detail subsequently, the set Y is employed to compute the 3-star count tensor T^{α_0} and sets A,B,C contain the leaves of the 3-stars under consideration. The roles of the sets can be interchanged to obtain the community membership vectors of all the sets, as described in Algorithm 1.

[8] To compute the modified moments G^{α_0}, and T^{α_0}, we need to know the value of the scalar $\alpha_0:=\sum_i\alpha_i$, which is the concentration parameter of the Dirichlet distribution and is a measure of the extent of overlap between the communities. We assume its knowledge here.

```
**Algorithm 1** $\{\hat{\Pi}, \hat{P}, \widehat{\alpha}\} \leftarrow$ LearnMixedMembership$(G, k, \alpha_0, N, \tau)$
**Input:** Adjacency matrix $G \in \mathbb{R}^{n\times n}$, $k$ is the number of communities, $\alpha_0 := \sum_i \alpha_i$, where
    $\alpha$ is the Dirichlet parameter vector, $N$ is the number of iterations for the tensor power
    method, and $\tau$ is used for thresholding the estimated community membership vectors,
    specified in (19) in assumption A5. Let $A^c := [n] \setminus A$ denote the set of nodes not in $A$.
**Output:** Estimates of the community membership vectors $\Pi \in \mathbb{R}^{n\times k}$, community connec-
    tivity matrix $P \in [0,1]^{k\times k}$, and the normalized Dirichlet parameter vector $\widehat{\alpha}$.
    Partition the vertex set $[n]$ into 5 parts $X$, $Y$, $A$, $B$, $C$.
    Compute moments $G^{\alpha_0}_{X,A}$, $G^{\alpha_0}_{X,B}$, $G^{\alpha_0}_{X,C}$, $\mathrm{T}^{\alpha_0}_{Y\to\{A,B,C\}}$ using (7) and (8).
    $\{\hat{\Pi}_{A^c}, \widehat{\alpha}\} \leftarrow$ LearnPartitionCommunity$(G^{\alpha_0}_{X,A}, G^{\alpha_0}_{X,B}, G^{\alpha_0}_{X,C}, \mathrm{T}^{\alpha_0}_{Y\to\{A,B,C\}}, G, N, \tau)$.
    Interchange roles[9] of $Y$ and $A$ to obtain $\hat{\Pi}_{Y^c}$.
    Define $\hat{Q}$ such that its $i$-th row is $\hat{Q}^i := (\alpha_0+1)\frac{\hat{\Pi}^i}{|\hat{\Pi}^i|_1} - \frac{\alpha_0}{n}\mathbf{1}^\top$.
    Estimate $\hat{P} \leftarrow \hat{Q}G\hat{Q}^\top$. {Recall that $\mathbb{E}[G] = \Pi^\top P \Pi$ in our model. We will show that
    $\hat{Q} \approx (\Pi^\dagger)^\top$.}
    Return $\hat{\Pi}, \hat{P}, \widehat{\alpha}$
```

Whitening: The whitening procedure attempts to convert the 3-star count tensor into an orthogonal symmetric tensor. Consider the k-rank singular value decomposition (SVD) of the modified adjacency matrix G^{α_0} defined in (7),

$$(|X|^{-1/2} G^{\alpha_0}_{X,A})^\top_{k-svd} = U_A D_A V_A^\top.$$

Define $\hat{W}_A := U_A D_A^{-1}$, and similarly define $\hat{W}_B$ and $\hat{W}_C$ using the corresponding matrices $G^{\alpha_0}_{X,B}$ and $G^{\alpha_0}_{X,C}$ respectively. Now define

$$\hat{R}_{A,B} := \frac{1}{|X|}\hat{W}_B^\top (G^{\alpha_0}_{X,B})^\top_{k-svd} \cdot (G^{\alpha_0}_{X,A})_{k-svd}\hat{W}_A, \tag{12}$$

and similarly define $\hat{R}_{AC}$. The whitened and symmetrized graph-moment tensor is now computed as

$$\mathrm{T}^{\alpha_0}_{Y\to\{A,B,C\}}(\hat{W}_A, \hat{W}_B\hat{R}_{AB}, \hat{W}_C\hat{R}_{AC}),$$

where T^{α_0} is given by (8) and the above describes a *multi-linear transformation* of the tensor.

Tensor power method: It can be shown that the whitening procedure yields a symmetric orthogonal tensor under exact moments. We now describe the tensor power method to recover components of a symmetric orthogonal tensor of the form

$$T = \sum_{i\in[r]} \lambda_i v_i \otimes v_i \otimes v_i = \sum_{i\in[r]} \lambda_i v_i^{\otimes 3}, \tag{13}$$

where r denotes the tensor rank and we use the notation $v_i^{\otimes 3} := v_i \otimes v_i \otimes v_i$, and the vectors $v_i \in \mathbb{R}^d$ are orthogonal to one another. Without loss of generality, we assume that

Procedure 1 $\{\hat{\Pi}_{A^c}, \widehat{\alpha}\} \leftarrow$ LearnPartitionCommunity($G^{\alpha_0}_{X,A}$, $G^{\alpha_0}_{X,B}$, $G^{\alpha_0}_{X,C}$, $\mathrm{T}^{\alpha_0}_{Y\to\{A,B,C\}}$, G, N, τ)

```
Compute rank-$k$ SVD: $(|X|^{-1/2}G^{\alpha_0}_{X,A})^\top_{k-svd} = U_A D_A V_A^\top$ and compute whitening matrices $\hat{W}_A := U_A D_A^{-1}$. Similarly, compute $\hat{W}_B, \hat{W}_C$ and $\hat{R}_{AB}, \hat{R}_{AC}$ using (12).
Compute whitened and symmetrized tensor $T \leftarrow \mathrm{T}^{\alpha_0}_{Y\to\{A,B,C\}}(\hat{W}_A, \hat{W}_B\hat{R}_{AB}, \hat{W}_C\hat{R}_{AC})$.
$\{\hat{\lambda}, \hat{\Phi}\} \leftarrow$TensorEigen$(T, \{\hat{W}_A^\top G_{i,A}^\top\}_{i\notin A}, N)$. $\{\hat{\Phi}$ is a $k\times k$ matrix with each columns being an estimated eigenvector and $\hat{\lambda}$ is the vector of estimated eigenvalues. $\{\hat{W}_A^\top G_{i,A}^\top\}_{i\notin A}$ is the set of initialization vectors and $N$ is the number of iterations.$\}$
$\hat{\Pi}_{A^c} \leftarrow$ Thres(Diag$(\hat{\lambda})^{-1}\hat{\Phi}^\top\hat{W}_A^\top G_{A^c,A}^\top$ , $\tau$) and $\hat{\alpha}_i \leftarrow \hat{\lambda}_i^{-2}$, for $i \in [k]$.
Return $\hat{\Pi}_{A^c}$ and $\hat{\alpha}$.
```

vectors $\{v_i\}$ are orthonormal in this case. In this case, each pair (λ_i, v_i), for $i \in [r]$, can be interpreted as an eigen-pair for the tensor T, since

$$T(I, v_i, v_i) = \sum_{j\in[r]} \lambda_j \langle v_i, v_j\rangle^2 v_j = \lambda_i v_i, \quad \forall i \in [r],$$

due to the fact that $\langle v_i, v_j\rangle = \delta_{i,j}$. Thus, the vectors $\{v_i\}_{i\in[r]}$ can be interpreted as fixed points of the map

$$v \mapsto \frac{T(I, v, v)}{\|T(I, v, v)\|}, \tag{14}$$

where $\|\cdot\|$ denotes the spectral norm (and $\|T(I, v, v)\|$ is a vector norm), and is used to normalize the vector v in (14). Thus, a straightforward approach to computing the orthogonal decomposition of a symmetric tensor is to iterate according to the fixed-point map in (14) with an arbitrary initialization vector. This is referred to as the tensor power iteration method. The simple power iteration procedure is however not sufficient to get good reconstruction guarantees under empirical moments. We make some modifications which involve *(i)* efficient initialization and *(ii)* adaptive deflation. The details are in the full version of the paper.

Reconstruction after tensor power method: When exact moments are available, estimating the community membership vectors Π is straightforward, once we recover all the stable tensor eigen-pairs, since $P \leftarrow (\Pi^\top)^\dagger\mathbb{E}[G|\Pi]\Pi^\dagger$. However, in case of empirical moments, we can obtain better guarantees with the following modification: the estimated community membership vectors $\hat{\Pi}$ are further subject to thresholding so that the weak values are set to zero. This yields better guarantees in the sparse regime of the Dirichlet distribution. In addition, we define $\hat{Q}$ such that its i^{th} row is

$$\hat{Q}^i := (\alpha_0 + 1)\frac{\hat{\Pi}^i}{|\hat{\Pi}^i|_1} - \frac{\alpha_0}{n}\mathbf{1}^\top,$$

based on estimate $\hat{\Pi}$, and the matrix $\hat{P}$ is obtained as $\hat{P} \leftarrow \hat{Q}G\hat{Q}^\top$. We subsequently establish that $\hat{Q}\hat{\Pi}^\top \approx I$, under a set of sufficient conditions outlined in the next section.

Improved support recovery estimates in homophilic models: A sub-class of community model are those satisfying *homophily*. Homophily is the tendency to form edges within the members of the same community, and has been posited as an important factor in community formation in social networks. We describe a post-processing method in Procedure 2 for models with community connectivity matrix P satisfying $P(i,i) \equiv p > P(i,j) \equiv q$ for all $i \neq j$. This yields a set of communities for each node where it has a significant presence, and we also rule out communities for every node where the presence is not strong enough.

Procedure 2 $\{\hat{S}\} \leftarrow \text{SupportRecoveryHomophilicModels}(G, k, \alpha_0, \xi, \hat{\Pi})$

Input: Adjacency matrix $G \in \mathbb{R}^{n\times n}$, k is the number of communities, $\alpha_0 := \sum_i \alpha_i$, where α is the Dirichlet parameter vector, ξ is the threshold for support recovery, corresponding to significant community memberships of an individual. Get estimate $\hat{\Pi}$ from Algorithm 1. Also asume the model is homophilic: $P(i,i) \equiv p > P(i,j) \equiv q$, for all $i \neq j$.

Output: $\hat{S} \in \{0,1\}^{n\times k}$ is the estimated support for significant community memberships.

```
Consider partitions $A, B, C, X, Y$ as in Algorithm 1.
Define $\hat{Q}$ on lines of definition in Algorithm 1, using estimates $\hat{\Pi}$. Let the $i$-th row for set $B$ be $\hat{Q}^i_B := (\alpha_0+1)\frac{\hat{\Pi}^i_B}{|\hat{\Pi}^i_B|_1} - \frac{\alpha_0}{n_B}\mathbf{1}^\top$. Similarly define $\hat{Q}^i_C$.
Estimate $\hat{F}_C \leftarrow G_{C,B}\hat{Q}_B^\top$, $\hat{P} \leftarrow \hat{Q}_C\hat{F}_C$.
if $\alpha_0 = 0$ (stochastic block model) then
  for $x \in C$ do
    Let $i^* \leftarrow \arg\max_{i\in[k]} \hat{F}_C(x,i)$ and $\hat{S}(i^*,x) \leftarrow 1$ and 0 o.w. {Assign community with maximum average degree.}
  end for
else
  Let $H$ be the average of diagonals of $\hat{P}$, $L$ be the average of off-diagonals of $\hat{P}$
  for $x \in C$, $i \in [k]$ do
    $\hat{S}(i,x) \leftarrow 1$ if $\hat{F}_C(x,i) \geq L + (H-L)\cdot\frac{3\xi}{4}$ and zero otherwise.{Identify large entries}
  end for
end if
Permute the roles of the sets $A, B, C, X, Y$ to get results for remaining nodes.
```

4. Sample Analysis for Proposed Learning Algorithm

4.1. Sufficient Conditions and Recovery Guarantees

It is easier to present the guarantees for our proposed algorithm for the special case, where all the communities have the same expected size, and the entries of the community connectivity matrix P are equal on diagonal and off-diagonal locations:

$$\widehat{\alpha}_i \equiv \frac{1}{k}, \qquad P(i,j) = p \cdot \mathbb{I}(i=j) + q \cdot \mathbb{I}(i \neq j), \quad p \geq q. \tag{15}$$

In other words, the probability of an edge according to P only depends on whether it is between two individuals of the same community or between different communities. The above setting is also well studied for stochastic block models ($\alpha_0 = 0$), allowing us to compare our results with existing ones. The results for general mixed membership models are available in the full version of the paper (Anandkumar et al., 2013).

[A1] Sparse regime of Dirichlet parameters: The community membership vectors are drawn from the Dirichlet distribution, $\mathrm{Dir}(\alpha)$, under the mixed membership model. We assume that $\alpha_i < 1$ for $i \in [k]$ $\alpha_i < 1$, which is the sparse regime of the Dirichlet distribution.

[A2] Condition on the network size: Given the concentration parameter of the Dirichlet distribution, $\alpha_0 := \sum_i \alpha_i$, we require that

$$n = \tilde{\Omega}(k^2(\alpha_0 + 1)^2), \tag{16}$$

and that the sets A, B, C, X, Y in the partition are $\Theta(n)$. Note that from assumption A1, $\alpha_i < 1$ which implies that $\alpha_0 < k$. Thus, in the worst-case, when $\alpha_0 = \Theta(k)$, we require[10] $n = \tilde{\Omega}(k^4)$, and in the best case, when $\alpha_0 = \Theta(1)$, we require $n = \tilde{\Omega}(k^2)$. The latter case includes the stochastic block model ($\alpha_0 = 0$).

[A3] Condition on edge connectivity: Recall that p is the probability of intra-community connectivity and q is the probability of inter-community connectivity. We require that

$$\frac{p - q}{\sqrt{p}} = \Omega\left(\frac{(\alpha_0 + 1)k}{n^{1/2}}\right) \tag{17}$$

The above condition is on the standardized separation between intra-community and inter-community connectivity (note that $\sqrt{p}$ is the standard deviation of a Bernoulli random variable). The above condition is required to control the perturbation in the whitened tensor (computed using observed network samples), thereby, providing guarantees on the estimated eigen-pairs through the tensor power method.

[A4] Condition on number of iterations of the power method: We assume that the number of iterations N of the tensor power method satisfies

$$N \geq C_2 \cdot \left(\log(k) + \log\log\left(\frac{p - q}{p}\right)\right), \tag{18}$$

for some constant C_2.

[A5] Choice of τ for thresholding community vector estimates: The threshold τ for obtaining estimates $\hat{\Pi}$ of community membership vectors in Algorithm 1 is chosen as

$$\tau = \begin{cases} \Theta\left(\frac{k\sqrt{\alpha_0}}{\sqrt{n}} \cdot \frac{\sqrt{p}}{p - q}\right), & \alpha_0 \neq 0, \quad (19) \\ 0.5, & \alpha_0 = 0, \quad (20) \end{cases}$$

For the stochastic block model ($\alpha_0 = 0$), since π_i is a basis vector, we can use a large threshold. For general models ($\alpha_0 \neq 0$), τ can be viewed as a regularization parameter and

[10] The notation $\tilde{\Omega}(\cdot), \tilde{O}(\cdot)$ denotes $\Omega(\cdot), O(\cdot)$ up to poly-log factors.

decays as $n^{-1/2}$ when other parameters are held fixed. We are now ready to state the error bounds on the estimates of community membership vectors Π and the block connectivity matrix P. $\hat{\Pi}$ and $\hat{P}$ are the estimates computed in Algorithm 1.

Recall that for a matrix M, $(M)^i$ and $(M)_i$ denote the i^{th} row and column respectively. We say that an event holds with high probability, if it occurs with probability $1-n^{-c}$ for some constant $c>0$.

Theorem 2 (Guarantees on Estimating P, Π) *Under A1-A5, we have w.h.p.*

$$\varepsilon_{\pi,\ell_1} := \max_i \|\hat{\Pi}^i - \Pi^i\|_1 = \tilde{O}\left(\frac{(\alpha_0+1)^{3/2}\sqrt{np}}{(p-q)}\right)$$

$$\varepsilon_P := \max_{i,j\in[n]} |\hat{P}_{i,j} - P_{i,j}| = \tilde{O}\left(\frac{(\alpha_0+1)^{3/2}k\sqrt{p}}{\sqrt{n}}\right).$$

The proofs are given in the full version of the paper (Anandkumar et al., 2013). The main ingredient in establishing the above result is the tensor concentration bound and additionally, recovery guarantees under the tensor power method. We now provide these results below.

Recall that $F_A := \Pi_A^\top P^\top$ and $\Phi = W_A^\top F_A \operatorname{Diag}(\widehat{\alpha}^{1/2})$ denotes the set of tensor eigenvectors under exact moments, and $\hat{\Phi}$ is the set of estimated eigenvectors under empirical moments. We establish the following guarantees.

Lemma 3 (Perturbation bound for estimated eigen-pairs) *Under the assumptions A1-A4, the recovered eigenvector-eigenvalue pairs $(\hat{\Phi}_i, \hat{\lambda}_i)$ from the tensor power method satisfies with high probability, for a permutation θ, such that*

$$\max_{i\in[k]} \|\hat{\Phi}_i - \Phi_{\theta(i)}\| \le 8k^{-1/2}\varepsilon_T, \qquad \max_i |\lambda_i - \widehat{\alpha}_{\theta(i)}^{-1/2}| \le 5\varepsilon_T, \tag{21}$$

The tensor perturbation bound ε_T is given by

$$\begin{aligned}\varepsilon_T &:= \left\|\mathrm{T}^{\alpha_0}_{Y\to\{A,B,C\}}(\hat{W}_A, \hat{W}_B\hat{R}_{AB}, \hat{W}_C\hat{R}_{AC}) - \mathbb{E}[\mathrm{T}^{\alpha_0}_{Y\to\{A,B,C\}}(W_A, \tilde{W}_B, \tilde{W}_C)|\Pi_{A\cup B\cup C}]\right\| \\ &= \tilde{O}\left(\frac{(\alpha_0+1)k^{3/2}\sqrt{p}}{(p-q)\sqrt{n}}\right),\end{aligned} \tag{22}$$

where $\|T\|$ for a tensor T refers to its spectral norm.

Stochastic block models ($\alpha_0 = 0$): For stochastic block models, assumptions A2 and A3 reduce to

$$n = \tilde{\Omega}(k^2), \qquad \zeta = \Theta\left(\frac{\sqrt{p}}{p-q}\right) = O\left(\frac{n^{1/2}}{k}\right). \tag{23}$$

This matches with the best known scaling (up to poly-log factors), and was previously achieved via convex optimization by Yudong et al. (2012) for stochastic block models. However, our results in Theorem 2 do not provide zero error guarantees as in (Yudong et al., 2012). We strengthen our results to provide zero-error guarantees in Section 4.1.1 below and thus, match the scaling of Yudong et al. (2012) for stochastic block models. Moreover, we also provide zero-error support recovery guarantees for recovering significant memberships of nodes in mixed membership models in Section 4.1.1.

Dependence on α_0: The guarantees degrade as α_0 increases, which is intuitive since the extent of community overlap increases. The requirement for scaling of n also grows as α_0 increases. Note that the guarantees on ε_π and ε_P can be improved by assuming a more stringent scaling of n with respect to α_0, rather than the one specified by A2.

4.1.1. Zero-error guarantees for support recovery

Recall that we proposed Procedure 2 as a post-processing step to provide improved support recovery estimates. We now provide guarantees for this method. We now specify the threshold ξ for support recovery in Procedure 2.

[A6] Choice of ξ for support recovery: The threshold ξ in Procedure 2 satisfies

$$\xi = \Omega(\varepsilon_P),$$

where ε_P is specified in Theorem 2. We now state the guarantees for support recovery.

Theorem 4 (Support recovery guarantees) *Assuming A1-A6 and* (15) *hold, the support recovery method in Procedure 2 has the following guarantees on the estimated support set $\hat{S}$: with high probability,*

$$\Pi(i,j) \geq \xi \Rightarrow \hat{S}(i,j) = 1 \quad \textit{and} \quad \Pi(i,j) \leq \frac{\xi}{2} \Rightarrow \hat{S}(i,j) = 0, \quad \forall i \in [k], j \in [n], \tag{24}$$

where Π is the true community membership matrix.

Thus, the above result guarantees that the Procedure 2 correctly recovers all the "large" entries of Π and also correctly rules out all the "small" entries in Π. In other words, we can correctly infer all the significant memberships of each node and also rule out the set of communities where a node does not have a strong presence.

The only shortcoming of the above result is that there is a gap between the "large" and "small" values, and for an intermediate set of values (in $[\xi/2, \xi]$), we cannot guarantee correct inferences about the community memberships. Note this gap depends on ε_P, the error in estimating the P matrix. This is intuitive, since as the error ε_P decreases, we can infer the community memberships over a large range of values.

For the special case of stochastic block models (i.e. $\lim \alpha_0 \to 0$), we can improve the above result and give a zero error guarantee at all nodes (w.h.p). Note that we no longer require a threshold ξ in this case, and only infer one community for each node.

Corollary 5 (Zero error guarantee for block models) *Assuming A1-A5 and* (15) *hold, the support recovery method in Procedure 2 correctly identifies the community memberships for all nodes with high probability in case of stochastic block models ($\alpha_0 \to 0$).*

Thus, with the above result, we match the state-of-art results of Yudong et al. (2012) for stochastic block models in terms of scaling requirements and recovery guarantees.

References

Edoardo M. Airoldi, David M. Blei, Stephen E. Fienberg, and Eric P. Xing. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 9:1981–2014, June 2008.

A. Anandkumar, D. P. Foster, D. Hsu, S. M. Kakade, and Y. Liu. Two svds suffice: Spectral decompositions for probabilistic topic modeling and latent dirichlet allocation, 2012a. arXiv:1204.6703.

A. Anandkumar, R. Ge, D. Hsu, S. M. Kakade, and M. Telgarsky. Tensor decompositions for latent variable models, 2012b.

A. Anandkumar, D. Hsu, and S.M. Kakade. A Method of Moments for Mixture Models and Hidden Markov Models. In *Proc. of Conf. on Learning Theory*, June 2012c.

A. Anandkumar, R. Ge, D. Hsu, and S. M. Kakade. A Tensor Spectral Approach to Learning Mixed Membership Community Models. *ArXiv 1302.2684*, Feb. 2013.

Sanjeev Arora, Rong Ge, Sushant Sachdeva, and Grant Schoenebeck. Finding overlapping communities in social networks: toward a rigorous approach. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, 2012.

Maria-Florina Balcan, Christian Borgs, Mark Braverman, Jennifer T. Chayes, and Shang-Hua Teng. I like her more than you: Self-determined communities. *CoRR*, abs/1201.4899, 2012.

Kamalika Chaudhuri, Fan Chung, and Alexander Tsiatas. Spectral clustering of graphs with general degrees in the extended planted partition model. *Journal of Machine Learning Research*, pages 1–23, 2012.

S.E. Fienberg, M.M. Meyer, and S.S. Wasserman. Statistical analysis of multiple sociometric relations. *Journal of the american Statistical association*, 80(389):51–67, 1985.

P. Gopalan, D. Mimno, S. Gerrish, M. Freedman, and D. Blei. Scalable inference of overlapping communities. In *Advances in Neural Information Processing Systems 25*, pages 2258–2266, 2012.

C. Hillar and L.-H. Lim. Most tensor problems are NP hard, 2012.

P.W. Holland, K.B. Laskey, and S. Leinhardt. Stochastic blockmodels: first steps. *Social networks*, 5(2):109–137, 1983.

T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51 (3):455, 2009.

F. McSherry. Spectral partitioning of random graphs. In *FOCS*, 2001.

G. Palla, I. Derényi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

T.A.B. Snijders and K. Nowicki. Estimation and prediction for stochastic blockmodels for graphs with latent block structure. *Journal of Classification*, 14(1):75–100, 1997.

J.A. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, 2012.

Y.J. Wang and G.Y. Wong. Stochastic blockmodels for directed graphs. *Journal of the American Statistical Association*, 82(397):8–19, 1987.

H.C. White, S.A. Boorman, and R.L. Breiger. Social structure from multiple networks. i. blockmodels of roles and positions. *American journal of sociology*, pages 730–780, 1976.

E.P. Xing, W. Fu, and L. Song. A state-space mixed membership blockmodel for dynamic network tomography. *The Annals of Applied Statistics*, 4(2):535–566, 2010.

Chen Yudong, Sujay Sanghavi, and Huan Xu. Clustering sparse graphs. In *Advances in Neural Information Processing Systems 25*, 2012.