An introduction to stochastic block models: drawbacks and a breakthrough

Liuqing Yang

May 10, 2018

1 Abstract

Real-world networks usually have community structure, that is, node are grouped into densely connected communities. Community detection is one of the most popular and best-studied research topics in network science and has attracted attention in many different fields, including computer science, statistics, social science, among others. Numerous approaches for community detection have been proposed in literature, from ad hoc algorithms to systematic-model-based approaches. The large number of available methods leads to a fundamental question: whether a certain method can provide consistent estimates of community labels. The stochastic block model (SBM) and its variants provide a convenient framework for the study of such problems. Although SBM has some theoretical advances such as consistency, before year of 2013, no model based computational method are useful for large network. This report will first introduce the stochastic block models and their theoretical properties, point out drawbacks in computation and then introduce a break through: Pseudo-likelihood method.

Key words: Stochastic Block Model, consistency, Pseudo-likelihood methods, EM algorithm.

Contents

2 Introduction

Network science is the study of networks (or graphs) as a representation of relations (called *links* or *edges*) between objects (called *vertices* or *nodes*). Networks have become one of the most common data structure. One famous example is the internet, which is the physical network, composed of computers, routers and modems linked by electronic, optical and wireless networking technologies. Other well-known examples include online social networks, gene regulatory networks, protein-protein interaction networks, food webs, among others. In the past decades, network science has drawn a lot of attention in many different branches of science and engineering, for example, computer science, physics, biology, social sciences, and economics. It is worth mentioning that network analysis has become an active research area in statistics. A number of probabilistic and statistical models have been proposed. Typical examples include the Erdős-Rényi random graph models, stochastic block models (SBMs) and others.
Most networks have community structure, that is, nodes are grouped into densely connected *communities* or *clusters*. Detection of such communities is one of the most popular research topics in network science. In this article, we adopt the most commonly used concept of community, that is, a community is a group of nodes with many links between themselves and fewer links to the rest of the network. Correspondingly, the goal of community detection is to partition the node set into overlapping or nonoverlapping cohesive communities. We focus on nonoverlapping community detection in this report.

Till now, methods for community detection can be basically classified into three categories. The first category consist of algorithm based methods such as hierarchical clustering which group node by a certain similarity measure. Methods in the second category are criterion-based methods, which optimizes some criteria over all possible network partitions. Methods in the third category are model-based. Such methods rely on fitting a probabilistic model for a network with community structure, in which the community labels are latent and to be identified. The best-studied model for community detection is the SBM, which plays a central role in the theoretical analysis of community detection. In this report, we will mainly focus on the introduction and discussion of the SBM and its variants. It is worth to note that there is no clear distinction among these categories. For instance, fitting a probabilistic model usually leads to a criterion to be optimized and the optimization eventually relies on an algorithm.

A fundamental question of community detection is whether a proposed

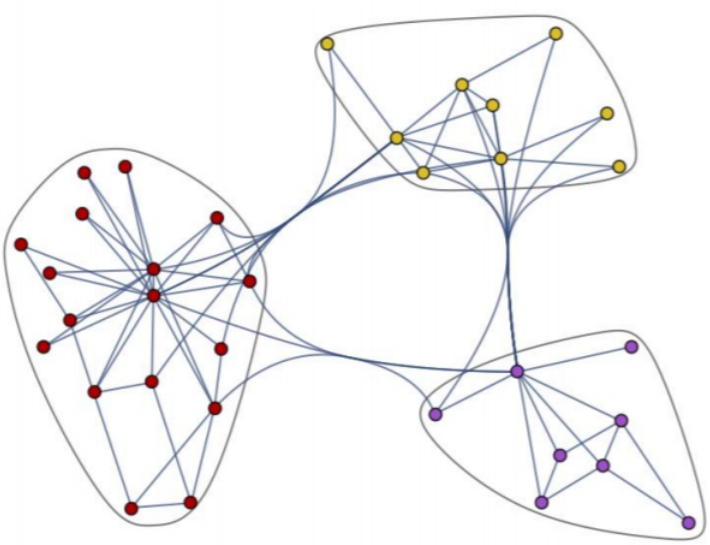

Figure 1: A example of network that has communities structures

method is able to correctly identify the community labels in principle. Or more precisely in statistical terminology, a fundamental theoretical question is whether a certain method can provide consistent estimates of community labels. Despite the conceptual similarity, community detection in networks is fundamentally different from clustering in multivariate data from a theoretical point of view.The structure of network data is unique. Unlike multivariate data, which are typically assumed to be independently and identically distributed, a network is represented by a single adjacency matrix and thereby no *replicates* in the usual sense are available. This unique data structure offers a great challenge in theoretical studies of community detection.

The SBM provided a natural framework for theoretical analysis of community detection. Under the SBM, many existing community detection methods are better understood and numerous new methods have been proposed and analyzed.These are the main focus of the current review article.The rest of this report is organized as follows. After introducing basic notations, we give the precise definition of the SBM. Next, we introduce some first results on consistency of community detection under the SBM and its variants. These results study the global optimization of these criteria over all possible label assignments. However, the global optimization of these criteria is in principle NP hard. Therefore, one computationally feasible method has been proposed, pseudo-likelihood method. This method has also been theoreti-

cally justified under the SBM and the corresponding results will be discussed in this report. In the last section, we will briefly discuss other research topics in community detection such as nodal covariates.

3 Stochastic Block Model

Before the invention of Stochastic Block Model, the most well-known random graph model is the Erdős-Rényi random graph models. At first people found it is consistent with some of the real world networks. However, because of this theoretical setting that every pair of two nodes in a network has the same probability to be connected, it failed to capture some topological properties of a network such as community structures. Based on that dilemma, Holland at 1983 gave the first introduce of the Stochastic Block Model which in the later, researchers found its decent capability to model the community structures for most real world networks.

3.1 Standard Stochastic Block Model

To introduce the statistical settings for the Stochastic Block model, let us begin by introducing basic notations. A network or a graph can be denoted by an ordered pair $N = (V, E)$, where V is the set of nodes and E is the set of edges. Without any loss of generality, we will assume $V = 1, ..., n$. A network with size n can be represented by an $n \times n$ adjacency matrix denoted as $A = [A_{ij}]$, where

$$A_{ij} = \begin{cases} 1 & \text{if there is an edge between node i and node j} \\ 1 & \text{otherwise} \end{cases}$$

Unless otherwise specified, we consider unweighted and undirected networks and thus A is a binary symmetric matrix. We assume that there is no self-loop in the network, i.e., $A_{ij} = 0$ for $i = 1, ..., n$.

We now formulate community detection and give the definition is to find a disjoint partition $V = V_1 \cup ... \cup V_K$, or equivalently node labels $e = (e_1, ..., e_n)$, where e_i is the label of node i and takes values in $\{1, 2, ..., K\}$. The SBM is perhaps the most commonly used model for representing a network with community structure. Under the SBM, a network is generated in two steps:

1. The true node labels $c = (c_1, ..., c_n)$ are drawn independently from Multinomial(1, π), where $\pi = (\pi_1, ..., \pi_K)$.

2. Given the labels c, the edge variables A_{ij} for $i < j$ are independently Bernoulli variables with

$$E[A_{ij}|c] = P_{c_i c_j}, \tag{1}$$

where $P = [P_{ab}]$ is a $K \times K$ symmetric matrix.

Before we proceed to discuss detection methods and theoretical results under the SBM, it is worth adding several remarks on the model itself:

- Firstly, the SBM can be understood as an analogy the Gaussian mixture model, for readers familiar with model-based clustering in multivariate analysis. But there is a crucial difference: The link probability for A_{ij} under the SBM depends on two community labels c_i and c_j, unlike the Gaussian mixture model. This 'two-dimensional' structure is the root cause of many computational challenges.

- Secondly, under the SBM, two nodes within a group are *stochastically equivalent* in terms of their link probabilities to other nodes. Intuitively speaking, two nodes within the same group play a similar role in the network. This leads back to the question of what is a community. As mentioned in the introduction, we treat community as a group of nodes with many links between themselves and fewer links to the rest of the network throughout this report. Thus to model communities in the usual sense, the SBM needs constraint on parameters that the within-group densities are larger than the cross-group densities, although many theoretical results do not require this constraint.

- Thirdly, the community labels c were treated as either random or deterministic in different literatures for their own technical conveniences. But in practical it makes little difference since c is unknown in either case (either latent random variables or unknown fixed parameters).

- Fourthly, note that the edge variables A_{ij} are independently given the labels and with $c_i = k$ and $c_i = l$, A_{ij} are identically distributed as Bernoulli(P_{kl}). Therefore, the SBM essentially assumes edge variables to be independently and identically distributed. This makes the SBM a convenient working model for studying asymptotic properties of community detection as size of the network goes into infinity.

3.2 Consistency for the Standard SBM

At the year of 2010, Bickel and Chen established a consistency framework for community detection. They developed general theory for checking the consistency of a large class of community detection criteria under the SBM as the number of nodes n goes and the number of communities K remains fixed.

For any label assignment e, let $O(e)$ be a $K \times K$ matrix with entries $\{O_{kl}(e)\}$ defined by

$$O_{kl}(e) = \sum_{1 \le i,j \le n} A_{ij} I\{e_i = k, e_j = l\},$$

$$n_k(e) = \sum_{i=1}^{n} I\{e_i = k\},$$

where I is the indicator function and define

$$L = \sum_{1 \le i,j \le n} A_{ij}.$$

For $k \neq l$, O_{kl} is the number of edges between communities k and l; O_{kk} is twice the number of edges within community k; $n_k(e)$ is the number of nodes in community k and L is the sum of all degrees in the whole network.

Define $f(e) = (n_1/n, n_2/n, ..., n_K/n)$ to be the fraction of nodes in each community and then a large class of community detection criteria can be written as the following general form up to a constant:

$$Q(e) = F(\frac{O(e)}{\mu_n}, \frac{L}{\mu_n}, f(e)),$$

where $\mu_n = E(L)$. This class include many graph cut methods and in this report we only discuss the profile likelihood criteria studied by Bickel and Chen of the SBM. Based on the former notations, the log-likelihood of A is:

$$\begin{aligned}
l(P|A) &= log(\prod_{i<j} P_{c_i c_j}^{A_{ij}} (1 - P_{c_i c_j})^{1 - A_{ij}}) \\
&= log([\prod_{1 \le i,j \le n} P_{c_i c_j}^{A_{ij}} (1 - P_{c_i c_j})^{1 - A_{ij}}]^{\frac{1}{2}}) \\
&= log([\prod_{1 \le k,l \le K} P_{kl}^{\sum A_{ij} I(e_i = k, e_j = l)} (1 - P_{kl})^{\sum I(e_i = k, e_j = l) - \sum A_{ij} I(e_i = k, e_j = l)}]^{\frac{1}{2}}) \\
&= \frac{1}{2} \sum_{1 \le k,l \le K} [O_{kl} log(P_{kl}) + (n_{kl} - O_{kl}) log(1 - P_{kl})],
\end{aligned}$$

where $n_{kl} = n_k n_l$ if $k \neq l$ and $n_{kk} = n_k(n_k - 1)$. In order to maximize the log-likelihood, we can first fix label assignment e and maximize it over P. By doing so, obtain the profile likelihood

$$Q_{SBM}(e) = \sum_{1 \leq k,l \leq K} n_{kl} \tau(\frac{O_{kl}}{n_{kl}}),$$

where $\tau(x) = xlogx + (1-x)log(1-x)$.

Remark 1. The community labels c_i are assumed to be fixed when the profile likelihood Q_{SBM} is derived. But c_i will be assumed to be random variables with Multinomial(1,π) in the later Theorem for consistency.

Let $\hat{c} = \arg\max_e Q_{SBM}(e)$. A natural necessary condition for consistency of $\hat{c}$ is that the 'limit' or 'population version' of $Q(e)$ should be maximized by the correct partition. Define $\lambda_n = \mu_n/n$ to be the average expected degree and $\rho_n = \mu_n/[n(n-1)]$ to be the expected graph density. Let R be a $K \times K$ matrix with entries $\{R_{ka}\}$ defined by

$$R_{ka} = \frac{1}{n} \sum_{i=1}^{n} I(e_i = k, c_i = a).$$

R_{ka} measures the fraction of nodes from community a but classified into community k. Define $S_{ab} = P_{ab}/\rho_n$ for $1 \leq a, b \leq K$. Note that S_{ab} is independent of n.

Bickel and Chen stated the following condition:
$F(RSR^T, 1, R\mathbf{1})$ is uniquely maximized over $\mathcal{R} = \{R : R \geq 0, R^T\mathbf{1} = \pi\}$ by $R = D(\pi)$, for all (π, S) in an open set Θ, where $\mathbf{1} = (1, ..., 1)^T$ and $D(\pi)$ is a diagonal matrix with π as its diagonal elements.

Remark 2. Despite its seemingly complicated form, the key condition is very natural, following the same principle of M-estimators.

Theorem 1. (Bickel and Chen)
Suppose F, S and π satisfy the above condition and some mild regularity conditions. Suppose $\lambda_n/logn \to \infty$. Then $\mathcal{P}[\hat{c} = c] \to 1$.

Remark 3. The result in Theorem 1 is called strong consistency in statistics literature or exact recovery in computer science literature. It requires no error in the estimated label vector with probability approaching to 1. During the

proof, Bickel and Chen also obtained the following result of weak consistency.

Theorem 2. (Bickel and Chen)
Under the same condition and suppose $\lambda_n \to \infty$, $\mathbb{P}(\frac{1}{n}\sum_{i=1}^{n}(\hat{c}_i \neq c_i) = 0) \to 1$

3.3 Degree-corrected Stochastic Block Model

The standard SBM implies that nodes within a community have the same expected degree. But high-degree nodes such as hub nodes do exist in many real-world networks and people found that it is very impractical to apply the standard SBM in this situation. To address this issue, Karrer and Newman in 2011 proposed the degree-corrected stochastic block model (DCSBM), which allows more variation among node degrees within a community.

Specifically, link probability in Eq. (1) was replaced with

$$E(A_{ij}|c) = \theta_i\theta_j P_{c_i c_j},$$

where parameter θ_i controls the degree of node i, under constraint $\sum_{i=1}^{n}\theta_i I(c_i = k) = 1$ for all $k = 1, ..., K$. This makes θ_i equal to the probability that an edge connected to the community to which node i belongs lands on itself. Thus given c and θ, the edges A_{ij} are independent Bernoulli random variables with $P(A_{ij} = 1|c, \theta) = \theta_i\theta_j P_{c_i c_j}$, where $P = [P_{ab}]$ is still a $K \times K$ symmetric matrix.

Zhao et al.in 2012 generalized the framework of Bickel and Chen and obtained a general theorem for consistency under the Degree-corrected Stochastic Block Model.

4 Pseudo-likelihood methods

Many community detection criteria have good theoretical properties under the framework of SBM. However, the optimization of these criteria, including the profile likelihood maximization, is a great challenge in practice. As a discrete optimization problem, finding global optimizers of these criteria requires the search over K^n possible assignments, which is computationally intractable especially for a network with large n.

The expectation-maximization (EM) algorithm for fitting the likelihood of SBM faces the same difficulty. Unlike fitting the Gaussian mixture model,

where the posterior probabilities of each cluster label can be calculated separately, the E-step for fitting SBM involves $KO(n^2)$ possible assignments. This is due to the 'two-dimensional' structure of networks as previously mentioned.

To overcome this issue, Amini et al. at 2013 proposed a scalable pseudo-likelihood method for fitting the SBM and DCSBM. He also proved consistency under the SBM with two communities.

4.1 Algorithms

4.1.1 Pseudo-likelihood (PL)

We adopt all the notation in the previous section and define a few more in order to introduce the method:

- Introduce an initial labelling vector $e = (e_1, ..., e_n)$.
- Define $b_{ik} = \sum_j A_{ij} 1(e_j = k)$ for $i = 1, ..., n$ and $k = 1, ..., K$. Let $b_i = (b_{i1}, ..., b_{iK})$.
- Let R be the $K \times K$ matrix with $R_{ka} = \frac{1}{n} \sum_{i=1}^{n} 1(e_i = k, c_i = a)$, $R_{k\cdot}$ = the k-th row of R.
- Let $P_{\cdot l}$ = the l-th column of P ($E(A_{ij}|c) = P_{c_i c_j}$ in SBM).
- Let $\lambda_{lk} = nR_{k\cdot}P_{\cdot l}$ and $\Lambda = \{\lambda_{lk}\}$.
- For each node i, conditional on labels $c = (c_1, ..., c_n)$ with $c_i = l$, let:
  - (A) $(b_{i1}, ..., b_{ik})$ are mutually independent.
  - (B) b_{ik}, a sum of independent Bernoulli variables, is approximately Poisson with mean λ_{ik}.

With true labels $\{c_i\}$ unknown, each b_i can be viewed as a mixture of Poisson vectors, identifiable as long as Λ has no identical rows.

By ignoring the dependence among $\{b_i, i = 1, ..., n\}$ using the Poisson assumption, treating $\{c_i\}$ as latent variables and setting $\lambda_l = \sum_k \lambda_{lk}$, we can write the pseudo log-likelihood as follows (up to a constant):

$$\ell_{PL}(\pi, \Lambda; \{b_i\}) = \sum_{i=1}^{n} log(\sum_{l=1}^{K} \pi_l e^{-\lambda_l} \prod_{k=1}^{K} \lambda_{lk}^{b_{ik}}).$$

Remark 1. Armini et al. made several approximations to obtain the above pseudo-likelihood:

1. The dependence among $\{b_i\}$ is ignored, which is reasonable since the dependence becomes very weak as n grows but K remains fixed.

2. Poisson approximation is used, which is also natural.

3. $L_{PL}(\{b_i\})$ is not a likelihood of the original adjacency matrix A anymore, but a likelihood of the block sums $\{b_i\}$, where b_i depend on the initial labeling e. Therefore, the performance of this method can be sensitive to the accuracy of the initial labeling.

A pseudo-likelihood estimate of (π, Λ) can then be obtained by maximizing $\ell_{PL}(\pi, \Lambda; \{b_i\})$. This can be done via the standard EM algorithm for mixture models, which alternates updating parameter values with updating probabilities of node labels. Once the EM converges, we update the initial block partition vector e to the most likely label for each node as indicated by EM and repeat this process for a fixed number of iteration T.

For any initial labeling e, let $n_k(e) = \sum_i 1(e_i = k)$, $n_{kl}(e) = n_k(e)n_l(e)$ if $k \neq l$, $n_{kk}(e) = n_k(e)(n_k(e) - 1)$ and $O_{kl}(e) = \sum_{i,j} A_{ij}1(e_i = k, e_j = l)$. We suppress the dependence on e whenever there is no ambiguity. The details of the algorithm can be summarized as follows.

The Pseudo-likelihood algorithm:
Initialize labels e and let $\hat{\pi}_l = n_l/n$, $\hat{R} = diag(\hat{\pi}_1, ..., \hat{\pi_K})$,$\hat{P_{lk}} = O_{lk}/n_{lk}$, $\hat{\lambda}_{lk} = n\hat{R_k}.\hat{P}_{\cdot l}$, $\hat{P} = \{\hat{P_{ik}}\}$ and $\hat{\Lambda} = \{\hat{\lambda_{lk}}\}$. Then repeat T times:

(1). Compare the block sums $\{b_{il}\}$ according to (2).

(2). (E-step) Using current parameter estimates $\hat{\pi}$ and $\hat{\Lambda}$, estimate probabilities for node labels by

$$\hat{\pi_{il}} = \mathbb{P}_{PL}(c_i = l|\mathbf{b_i}) = \frac{\hat{\pi_l} \prod_{m=1}^{K} exp(b_{im} log\hat{\lambda_{lm}} - \hat{\lambda_{lm}})}{\sum_{k=1}^{K} \hat{\pi_l} \prod_{m=1}^{K} exp(b_{im} log\hat{\lambda_{lm}} - \hat{\lambda_{lm}})}$$

(3). (M-step) Given label probabilities, update parameter values as follows:

$$\hat{\pi_l} = \frac{1}{n} \sum_{i=1}^{n} \hat{\pi_{il}}, \quad \hat{\lambda_{lk}} = \frac{\sum_i \hat{\pi_{il}} b_{ik}}{\sum_i \hat{\pi_{il}}}$$

(4). Return to step2 unless the parameter estimates converge

(5). Update labels by $e_i = \arg\max_l \hat{\pi_{il}}$ and return to step1.

(6). Update $\hat{P}$ by $\hat{P_{lk}} = \frac{\sum_{i,j} A_{ij}\hat{\pi_{il}}\hat{\pi_{ik}}}{\sum_i 1(e_i=l)\sum_j 1(e_j=k)}$.

Armini et al. found that empirically that the algorithm above is more stable and converges faster. They found in general, only need a few label updates until convergence and even using $(T = 1)$ (one-step label update) gives the reasonable results with a good initial value. The choice of the initial value of e, on the other hand, can be important; see more details on this in Armini et al. 2013 Section 2.3.

4.1.2 Pseudo-likelihood conditional on node degrees (CPL)

As discussed in Section 3.3, for networks with hub nodes or those with substantial degree variability within communities, the block model can provide a poor fit, essentially dividing the nodes into low-degree and high-degree groups. This has been both observed empirically and supported by theory. Thus the degree-corrected block model was designed to cope this situation and writing out a pseudo-likelihood that lends itself to an EM-type optimization is more complicated. However, Armini et al. found there is a simple alternative: consider the pseudo-likelihood conditional on the observed node degrees. Whether these degrees are similar or not will not then matter and the fitted parameters will reflect the underlying block structure rather than the similarities in degrees.

The conditional pseudo-likelihood is again based on a simple observation:

- If random variables X_k are independent Poisson with means μ_k, their distribution conditional on $\sum_k X_k$ is multinomial.

Applying this observation to the variables $(b_{i1}, ..., b_{iK})$, we have that their distribution conditional on labels c with $c_i = l$ and the node degree $d_i = \sum_k b_{ik}$, the distribution of $(b_{i1}, ..., b_{iK})$ is multinomial with parameters $(d_i; \theta_{l1}, ..., \theta_{lk})$, where $\theta_{lk} = \frac{\lambda_{lk}}{\lambda_l}$ The conditional log pseudo-likelihood(up to a constant) is:

$$\ell_{CPL}(\pi, \Lambda; \{b_i\}) \propto \sum_{i=1}^{n} log(\sum_{i=1}^{K} \pi_l \prod_{k=1}^{K} \theta_{lk}^{b_{ik}}).$$

and the parameter can be obtained by maximizing this function via the EM algorithm for mixture models, as before. We again repeat the EM for a fixed number of iterations, updating the initial partition vector after the EM has converged. The algorithm is then the same as that for unconditional pseudo-likelihood, with steps 2 and 3 replaced by:

(2'). (E-step) Using current parameter estimates $\hat{\pi}$ and $\hat{\theta_{lk}}$, estimate probabilities for node labels by

$$\hat{\pi_{il}} = \mathbb{P}_{PL}(c_i = l|\mathbf{b_i}) = \frac{\hat{\pi}_l \prod_{m=1}^{K} \hat{\theta}_{lm}^{b_{im}}}{\sum_{k=1}^{K} \hat{\pi}_l \prod_{m=1}^{K} \hat{\theta}_{lm}^{b_{im}}}$$

(3'). (M-step) Given label probabilities, update parameter values as follows:

$$\hat{\pi_l} = \frac{1}{n} \sum_{i=1}^{n} \hat{\pi_{il}}, \quad \hat{\theta_{lk}} = \frac{\sum_i \hat{\pi_{il}} b_{ik}}{\sum_i \hat{\pi_{il}} d_i}$$

4.2 Consistency results

Armini et al. proved the weak consistency (recall Theorem 2 in Section 3.2) of the estimator from one-step EM of CPL for $K = 2$ under the SBM. True community labels c are treated as fixed parameters. For simplicity, we only present the result for balanced communities, i.e., each community contains $m = n/2$ nodes. Assume the link probability matrix P has the form :

$$P = \frac{1}{m} \begin{bmatrix} a & b \\ b & a \end{bmatrix}. \tag{2}$$

Then starts from some initial estimate $\hat{a}$, $\hat{b}$ and $\hat{\pi} = (\hat{\pi_1}, \hat{\pi_2})$ of parameters a, b and π, together with an initial labelling e and output the label estimates:

$$\hat{c}_i(e) = \arg\max_{k \in \{1,2\}} \{log\hat{\pi}_k + \sum_{l=1}^{2} b_{il}(e) log\hat{\theta}_{kl}(e)\}, \quad i = 1, ..., n \tag{3}$$

where $\hat{\theta}_{kl}$ are the elements of the matrix obtained by row normalization of $\hat{\Lambda} = [nR(e)\hat{P}]$. Here $R = R(e)$ is the $K \times K$ confusion matrix defined in Section 4.1.1 and $\hat{P}$ is given by (2), depending on the model with a and b replaced with their estimates $\hat{a}$ and $\hat{b}$. Also assume that the initial labeling e is balanced and it matches exactly γm labels in community 1.

Theorem. (Theorem 2 in Armini et al. 2013)
The one-step EM estimator of CPL is weakly consistent, i.e. $\mathbb{P}(\frac{1}{n}\sum_{i=1}^{n}(\hat{c}_i(e) \neq c_i) = 0) \to 1$, *under some mild regularity conditions and the following main assumptions:*

(C1) $\gamma \neq 1/2$;

(C2) $(\hat{a} - \hat{b})(a - b) > 0$;

(C3) $(a - b)^2/(a + b) \to \infty$.

All these assumptions are intuitive and very mild. Condition (C1) only requires the initial labeling better than random guessing. Condition (C2) means that the estimates $(\hat{a}, \hat{b})$ should have the same ordering as true parameters (a, b). It is also easy to check that $\lambda_n \to \infty$ implies (C3). On the other hand, it is worth noting that Theorem only guarantees consistency for the case of two communities. The proof, details in Armini et al. 2013, is already highly technical and relies on advanced probability tools. It may be quite challenging to prove or even formulate the theorem for the general case.

4.3 Numeric results

Here we investigate the performance of both the unconditional and conditional pseudo-likelihood algorithms on simulated networks, as well as that of spectral clustering. We simulate two scenarios, one from the regular stochastic block model and one from the degree-corrected block model to assess the performance in the presence of hub nodes. Throughout this section, we fix $n = 1000$, $K = 2$ and $\pi = (1/2, 1/2)$. Conditional on the labels, the edges are generated as independent Bernoulli variables with probabilities proportional to $\theta_i \theta_j P_{ij}$. The parameters θ_j are drawn independently from the distribution of Θ with $\mathbb{P}(\Theta = 0.2) = \rho$, $\mathbb{P}(\Theta = 1) = 1 - \rho$. We consider two values of $\rho : \rho = 0$, which corresponds to the regular block model; $\rho = 0.5$, which corresponds to a network where 50% of the nodes can be viewed as hub nodes.

The matrix P is constructed as follows:

$$P = \frac{\lambda}{(n-1)(\pi^T P^{(0)} \pi)(E(\Theta))^2} P^{(0)}.$$

1. λ is the overall expected network degree which control the P and varies from 1 to 12.

2. Let
$$P^{(0)} = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix},$$
which allows the within community connections is 9 times as many as between community connections.

3. From our simulation setting $E(\Theta) = 1 - 0.8\rho$.

To compare our results to the true labels, we used the proportion of correctly clustered labels. All figures show the performance of the following three methods: Spectral clustering (SC), unconditional pseudo-likelihood with initial label e given by spectral clustering (PL.SC), conditional pseudo-likelihood with initial label e given by spectral clustering (CPL.SC). The number of outer iterations for PL.SC and CPL.SC is set to $T = 20$.

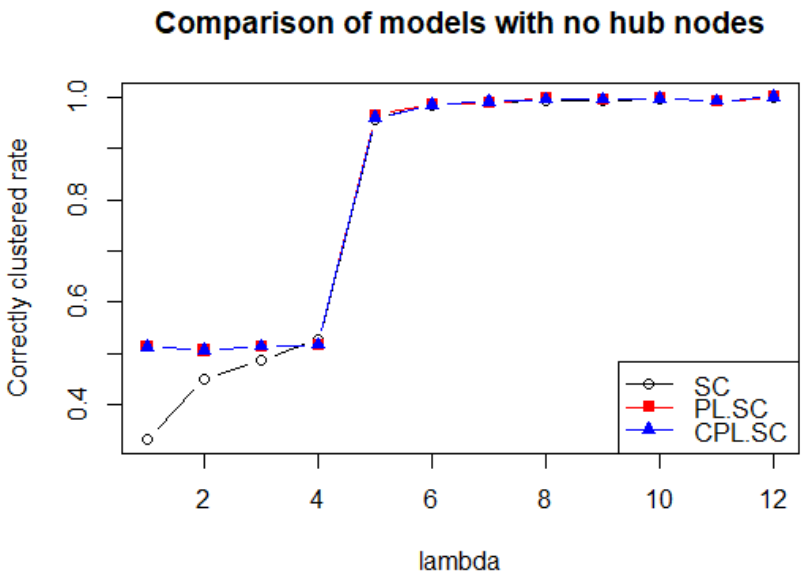

Figure 2: The proportion of correctly clustered labels as a function of λ when $\rho = 0$, i.e. no hub nodes

Figure 2 show results on estimating the node labels with varying λ without hub node. Generally, larger λ, i.e. denser network, make the clustering easier as we expect. However, when the network is not dense enough or even sparse, the performance of both PL.SC and CPL.SC is better than the spectral clustering. This observation is consistent with the numeric results in Armini et al 2013. They actually proposed PL and CPL to, on the other hand, overcame the poor performance of existing algorithm based method such as SC when the network is large and sparse. No difference observed between PL.SC and CPL.SC.

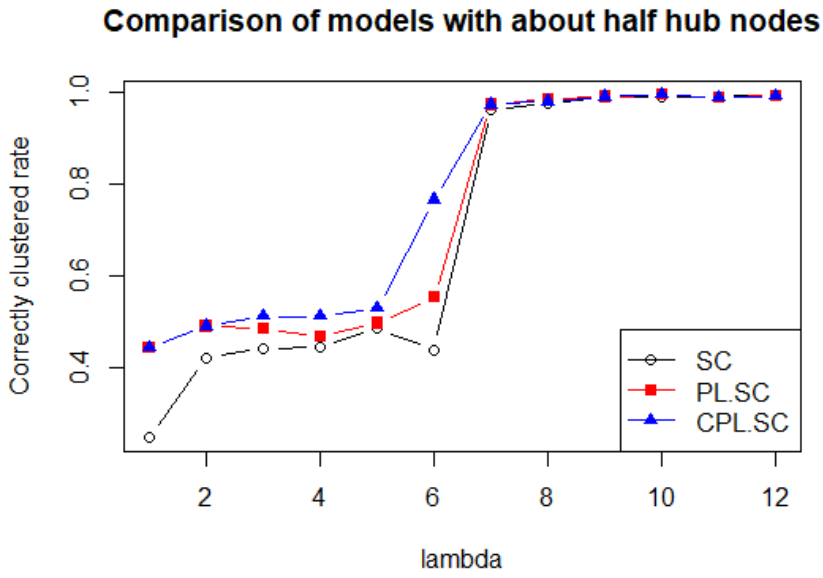

Figure 3: The proportion of correctly clustered labels as a function of λ when $\rho = 0.5$, i.e. half of the nodes are hub nodes

This time, as we can expected, the CPL.SC becomes the best one when hub nodes exist and it also shows the necessity of the idea of "Degree corrected". Although PL.SC still performs better than SC, the superiority is not obvious except when network is sparse. Still when the network is sense enough, three models all perform very well because of enough information.

n	SC	PL.SC	CPL.SC
50	0.015	0.0682	0.0521
100	0.026	0.1082	0.1153
1000	4.8038	5.4545	5.4094
5000	510.65	524.33	521.73

Table 1: Running time in seconds for three models when number of nodes n increases

As stated in the beginning, one main aim of Pseudo-likelihood method is to overcome the computing disadvantages of SBM. Table1 shows that the time consuming of either unconditional or conditional pseudo-likelihood method is acceptable and not much bigger than that of spectral clustering.

5 Discussion

Traditional community detection approaches only use the adjacency matrix, i.e. the network itself as the input. However, additional information on the nodes is usually available in addition to network topology. Thus a natural question is how or whether we can improve community detection by using node features, when presumably these features are correlated to community structure. Thus a particular challenge in community detection with nodal covariates is how to assess whether or not covariates are correlated with the community structure induced by the adjacency matrix. Sometimes, covariates and the network showed different community structures. Even when they are correlated, it is not clear whether combining them is necessarily better than using only one source and therefore more research can be conducted for this question.

References

- Amini, A. A., Chen, A., Bickel, P. J., & Levina, E. (2013). Pseudo-likelihood methods for community detection in large sparse networks. *The Annals of Statistics,* 41(4), 2097-2122.
- Bickel, P. J., & Chen, A. (2009). A nonparametric view of network models and Newman–Girvan and other modularities. *Proceedings of the National Academy of Sciences,* 106(50), 21068-21073.
- Karrer, B., & Newman, M. E. (2011). Stochastic blockmodels and community structure in networks. *Physical review E,* 83(1), 016107.
- Zhao, Y., Levina, E., Zhu, J. (2012). Consistency of community detection in networks under degree-corrected stochastic block models. *The Annals of Statistics,* 40(4), 2266-2292.
- Zhao, Y. (2017). A survey on theoretical advances of community detection in networks. *Wiley Interdisciplinary Reviews: Computational Statistics,* 9(5).