Active Community Detection with Maximal Expected Model Change

Dan Kushnir
NOKIA Bell Laboratories USA

Benjamin Mirabelli
Princeton University

Abstract

We present a novel active learning algorithm for community detection on networks. Our proposed algorithm uses a Maximal Expected Model Change (MEMC) criterion for querying network nodes label assignments. MEMC detects nodes that maximally change the community assignment likelihood model following a query. Our method is inspired by detection in the benchmark Stochastic Block Model (SBM), where we provide sample complexity analysis and empirical study with SBM and real network data for binary as well as for the multi-class settings. The analysis also covers the most challenging case of sparse degree and below-detection-threshold SBMs, where we observe a super-linear error reduction. MEMC is shown to be superior to the random selection baseline and other state-of-the-art active learners.

1 Introduction

Community detection, or clustering on networks, is a fundamental problem in a broad range of disciplines, from the study of biological and social networks to the classification of non-graphical data sets via the construction of pairwise similarity graphs [Tang and Liu(2010)]. However, perfectly recovering each element's community by only observing the given graphical data is shown to be statistically impossible for many networks of interest. Therefore, a growing area of research has been focused on the development of semi-supervised community detection algorithms [Allahverdyan et al.(2010), Eaton and Mansbach(2012), Kanade et al.(2016), Zhang et al.(2014)]. More recently active learning [Settles(2010)] has been introduced to the task of community detection (e.g. [Cheng et al.(2014), Gadde et al.(2016), Moore et al.(2011)]). Active learning, in this setting, allows to use a minimal and intelligently selected set of nodes to be labelled in order to improve community detection. Active learning is especially beneficial when labeling information and training are hard or expensive to obtain.

In this paper we focus on using active learning to solve both the problem of 'detection' and 'recovery' of communities: where 'detection' is related to finding community assignments that have a non-trivial correlation with the true assignments, and recovery addresses the assignments to be correct with vanishing error probability. In the context of detection [Decelle et al.(2011), Mossel(2015)] have shown that for benchmark random networks generated by the stochastic block model (SBM) [Holland et al.(1983)], there exists a fundamental 'detection threshold' for the unsupervised setting. These results prove that when the signal to noise ratio (SNR) of SBM-generated graphical data is below the 'detection' threshold (SNR<1) then it is statistically impossible for any strictly graph-based algorithm to predict community assignments with better accuracy than random chance [Mossel(2015), Abbe(2017)]. Recent work on clustering SBM networks with random label information [Saad and Nosratinia(2018), Mossel and Xu(2016), Kanade et al.(2016)] (aka semi-supervised) has shown detection is possible below the threshold. We note [Kanade et al.(2016)] for the challenging case of sparse degree networks, where equivalence between label propagation in broadcast-trees and SBM is drawn, also covering the SNR<1 case.

Active learning in the context of community detection is a new and pioneering area, in particular in the context of SBMs. So far, only a handful of active learning strategies have been introduced to the field of community-detection. In the context of SBM we note entropy-based selection criterion of [Moore et al.(2011)], which computes the difference between a node's uncertainty and its correlation with

Proceedings of the 23[rd]International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108.

other graph nodes in order to select the best node to query. The computationally demanding Gibbs sampling is used to derive the conditional distribution of node labels in SBM. In [Gadde et al.(2016)] the authors address the perfect recovery problem in the SBM's logarithmic degree regime, which uses sampling of nodes whose neighbor's labels suggest the highest disagreement on their current assignment. The sampling is done once from the initial label approximation, and there is no update of the labeling after each query\sample. Thus, hypothesis updates, which prevent redundant queries, are not exploited. Other network\graph based active learning methods in other contexts than SBM involve criteria such as 'representative' sampling [Leng et al.(2013)], uncertainty sampling [Kushnir(2014)], and a combination of thereof [Cheng et al.(2014), Yang et al.(2015)]. [Cheng et al.(2014)] targets edge queries, however, node queries typically provide more information.

In this paper we construct a novel active learner that embeds a semi-supervised community-detection algorithm based on optimizing the likelihood function. Specifically in the case of SBM, we find the maximum-likelihood (ML) community labeling given a SBM-generated graph and a subset of already known community labels. In this sense our work takes the SDP solution of [Goemans and Williamson(1995)] for the unsupervised ML problem one step further into the semi-supervised domain, by using the labeling information as constraints. Our active learner utilizes the ML statistical framework in order to query nodes that Maximize the Expected Model Change (MEMC) of the likelihood model given the graph, labels, and the hypothesized query information. The MEMC criterion is comprised of optimizing two components simultaneously: the overall change in the global likelihood model for each query outcome, and the likelihood of the query label given the current model. Model change has been used for active learning in other frameworks (e.g. [Freytag et al.(2014), Settles(2010), Vezhnevets et al.(2012)]) but is novel for ML and community detection.

MEMC's sample complexity analysis marks its advantage over the baseline Random selection criterion in both below and above SNR-threshold networks and for different average node degree regimes. In particular, the analysis proves MEMC's preference to correct erroneous assignments, and bounds its query search range, while the Random sampling range is unbounded. The theoretical analysis is validated with an empirical study of the sample complexity. In the below SNR regime, our analysis and simulations reveal a surprisingly fast super-linear error reduction phase comprised of multiple nodes label corrections following a query. The super-linear phase is followed by a linear reduction phase that lasts until full recovery and where MEMC has a clear advantage over Random. We conclude our work with numerical experiments with SBM networks which validated our analysis, as well as experiments with real social networks showing a clear advantage of MEMC over the baseline and other state-of-the-art active learning.

2 SBM and the Likelihood Function

We start with definitions related to the SBM and derive new results pertaining to its likelihood.

Definition 1. *The Stochastic Block Model (SBM) with parameters n, r, a and b defines a random graph ensemble on n nodes. Each node is assigned a label, uniformly at random, from a discrete set of r labels. If node i and node j are assigned the same label there exists an edge between them with independent probability $p = \frac{a}{n}$ and if node i and node j are assigned different labels there exists an edge between them with independent probability $q = \frac{b}{n}$.*

When the number of nodes in each community is equal we refer to the model as the *Symmetric SBM* (SSBM).

Definition 2. *(Modified Adjacency Matrix) Given a graph G on n vertices and parameters p and q, let $M = M(G,p,q)$ be the $n \times n$ 'modified' adjacency matrix where $M_{ij} = \log \frac{p}{q}$ if G has an edge between node i and node j and $M_{ij} = \log\left(\frac{1-p}{1-q}\right)$ if G does not have an edge between node i and node j. Let $SBM(n,r,p,q)$ represent the probability distribution over matrices M.*

For completion we provide the fundamental definition of the detection Signal-to-Noise ratio for SBMs [Decelle et al.(2011)] and the seminal results on community detection of [Massoulie(2014), Mossel et al.(2015)] for $r = 2$:

Definition 3. *The Signal-to-Noise-Ratio (SNR) of $SBM(n,r,\frac{a}{n},\frac{b}{n})$ is defined as*

$$SNR \doteq \frac{(a-b)^2}{r(a+(r-1)b}. \tag{1}$$

Theorem 1. *(Detection Threshold) [Massoulie(2014), Mossel et al.(2015)]: for r=2, if $(a-b)^2 > 2(a+b)$ then one can a.a.s. find a bisection which is positively correlated with $SBM(n,2,\frac{a}{n},\frac{b}{n})$.*

Definition 4. *(Vector Labels). Let $\triangle_r$ be a set of r unit-vectors representing the vertices of an $(r-1)$-simplex and let $\triangle_r^n$ be the set of all n-tuples of such vectors. Thus, if $\left[\begin{smallmatrix} v_i \\ v_j \end{smallmatrix}\right] \in \triangle_r^2$, then the inner-product*

$$\langle v_i, v_j \rangle = \begin{cases} -\frac{1}{r-1} & \text{if } v_i \neq v_j \\ 1 & \text{if } v_i = v_j. \end{cases}$$

From now on we associate any discrete labeling over r communities with vectors in the set $\triangle_r$. Thus, we refer to a complete labeling over n nodes as $X \in \triangle_r^n$, where $X_i \in \triangle_r$ is the i^{th} node's label and corresponds to the i^{th} row of X. In Theorem 2 we derive the unnormalized probability of any discrete community labeling assignment $x \in \triangle_r^n$ as a monotonic and convex function of the quadratic-form $Tr(x^T M x)$.

Theorem 2. *Let $X \in \triangle_r^n$ be a discrete labeling assignment and let $\boldsymbol{M} \sim SBM(n, r, p, q)$. Then*

$$\mathbb{P}[\boldsymbol{X} = X | \boldsymbol{M} = M] \propto e^{\frac{r-1}{2r} Tr(X^T M X)}. \tag{2}$$

Corollary 1. *Let $X_U \in \triangle_r^{n-k}$ be a discrete community labeling assignment for a set of $n-k$ unsupervised nodes and let $X_L \in \triangle_r^k$ be the true labeling assignment for a subset of k supervised nodes. Let $\boldsymbol{M} \sim SBM(n, r, p, q)$ where the last k rows correspond to the k supervised nodes. Then,*

$$\mathbb{P}[\boldsymbol{X}_U = X_U | \boldsymbol{M} = M, \boldsymbol{X}_L = X_L] \propto e^{\frac{r-1}{2r} Tr(\left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right]^T M \left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right])}.$$

3 Semi-Supervised

For the problem of semi-supervised community-detection, we seek the discrete labeling assignment $X_U \in \triangle_r^{n-k}$ that maximizes $\mathbb{P}[\mathbf{X}_U = X_U | \mathbf{M} = M, \mathbf{X}_L = X_L]$ (k being the the number of labeled nodes). From Corollary 1, this is equivalent to finding the labeling assignment $X_U \in \triangle_r^{n-k}$ that maximizes $Tr(\left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right]^T M \left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right])$. However, in general, exactly finding this discrete ML labeling is exponentially hard and can naively take up to $O(r^{n-k})$-time. Thus, we relax the optimization problem into a SDP and find an 'approximate' ML labeling in the relaxed domain.

While SDP has been used in the unsupervised case to solve community detection problems (see references in [Abbe(2017)]), we present here a new formulation for the Semi-supervised case using the labeled set as a set of constraints in the original SDP. Specifically, given the graph $\mathbf{M} = M$ and labeling $\mathbf{X}_L = X_L$, the SDP of our semi-supervised algorithm maximizes the convex function $Tr(\left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right]^T M \left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right])$ over the 'relaxed' domain $X_U \in \mathbb{R}^{(n-k)\times(n-k+r)}$ and $\|X_i\|_2 = 1$ for every i. We find the optimal assignment for X_U in this relaxed domain by factoring the solution $\mathbb{X} = \left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right]\left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right]^T$ of the following Semi-Definite Program:

$$\begin{aligned} \text{SDP}(M, X_L): \max_{\mathbb{X}} \ & \text{Tr}(M\mathbb{X}) \text{ such that} \\ & \mathbb{X} \succeq 0, \\ & \mathbb{X}_{ii} = 1 \text{ for } 1 \le i \le n; \\ & \mathbb{X}_{ij} = 1 \text{ if } X_{L\_i} = X_{L\_j}, \\ & \mathbb{X}_{ij} = -\frac{1}{r-1} \text{ if } X_{L\_i} \ne X_{L\_j}; \end{aligned} \tag{3}$$

where X_{U_i} or X_{L_i} is just another way of writing X_i while simultaneously specifying if node i is currently in the unlabeled or labeled set.

We define the output of $\text{SDP}(M, X_L)$ to be the factorized matrix $X = \left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right]$ rotated so that the vectors X_L line up with their correct corresponding vectors in $\triangle_r$. This SDP can be solved efficiently with programs such as Manopt [Bandeira et al.(2016)]. The factorization $\mathbb{X} = \left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right]\left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right]^T$ can be found efficiently with Cholesky Decomposition and $X = \left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right]$ has a unique rotated solution so long as there is at least one supervised label from each community (if not see Remark 1 below). From the constraints $\mathbb{X}_{ii} = 1$, each X_i is a unit vector and referred to as the 'vector-label' for node i. To complete the semi-supervised algorithm we recover a discrete labeling by assigning each vector-label in X_U to the closest corresponding vector-label in $\triangle_r$.

Remark 1. *If X_L contains less than $r-1$ distinct vectors, then rotating X such that X_L aligns with its corresponding vectors in $\triangle_r$ no longer produces a unique solution with respect to $\triangle_r$. In this case we must use another algorithm to find the best-fit simplex for our data. We present one such algorithm in the supplemental material.*

The semi-supervised algorithm (Fig. 1) follows a relax-and-round procedure where the relaxed SDP (3) is solved, followed by fitting a simplex to the SDP-output and rounding each SDP vector-label to its closest best-fit-simplex vector. Note that the function $unique(X)$ outputs the set of unique labels in X.

Semi-Supervised(M, X_L, r)

Input: M: adjacency matrix, X_L: labeled set, r: number of communities
Output: $X \in \triangle_r^n$: complete labeling
1. $X' = \text{SDP}(M, X_L)$
2. If $|\text{unique}(X_L)| < r$
   $\triangle_r = \text{bestFitSimplex}(X')$
   else
   $\triangle_r = \text{unique}(X_L)$
3. For $i = 1$ to $n - k$
   $X_{U\_i} = \underset{X_j \in \triangle_r}{\text{argmax}} \ X'_{U\_i} X_j^T$
4. $X = \left[\begin{smallmatrix} X_U \\ X_L \end{smallmatrix}\right]$

Figure 1: The *Semi-Supervised* algorithm.

4 Active Learning

Our novel active learning strategy employs a querying criterion that is based on Maximal Expected Model Change (MEMC). In general, this strategy selects the

unlabeled data point, $q \in U$, that if labeled is expected to cause the greatest change to some chosen model Φ with respect to some chosen norm T. Built on this idea, a tractable approximation of the distribution $\mathbb{P}[\mathbf{X}_q = X_q|\mathbf{M} = M, \mathbf{X}_L = X_L]$ must be computed for all $q \in U$ and $X_q \in \triangle_r$, and a model Φ is to be chosen.

4.1 Expectation

In computing the *expected* model change, the expectation must be taken with respect to each node's likelihood distribution over possible label assignments. However, exactly computing this distribution, $\mathbb{P}[\mathbf{X}_i = X_i|\mathbf{M} = M, \mathbf{X}_L = X_L]$, is an exponentially hard problem with complexity $O(r^{n-k})$ in general. Hence, we use the classical strategy of ML-approximation [Liu(2012)], to approximate this distribution by

$$\mathbb{P}[\mathbf{X}_i = X_i|\mathbf{M} = M, \mathbf{X}_L = X_L, \mathbf{X}_{U_{\neg i}} = X_{U_{ML}}], \quad (4)$$

where $\mathbf{X}_{U_{\neg i}}$ is the set of unknown node labels excluding node i, and $X_{U_{ML}}$ is the ML-labeling. For any labeling $\mathbf{X}_{U_{\neg i}} = X_{U_{\neg i}}$ one can compute the above conditional probability with Lemma 1:

Lemma 1. *Let $X_{U_{\neg i}} \in \triangle_r^{(n-k-1)}$ be a discrete community labeling assignment, let $\boldsymbol{M} \sim SBM(n, r, p, q)$ and let $\boldsymbol{M}_i$ be the i^{th} row of $\boldsymbol{M}$ without the (i, i) entry. Then, for any label $X_i \in \triangle_r$,*

$$\mathbb{P}[\boldsymbol{X}_i = X_i|\boldsymbol{M} = M, \boldsymbol{X}_L = X_L, \boldsymbol{X}_{U_{\neg i}} = X_{U_{\neg i}}] = \frac{e^{\frac{r-1}{r}(M_i\left[\begin{smallmatrix} X_{U_{\neg i}} \\ X_L \end{smallmatrix}\right]X_i^T)}}{\sum_{X_j \in \triangle_r} e^{\frac{r-1}{r}(M_i\left[\begin{smallmatrix} X_{U_{\neg i}} \\ X_L \end{smallmatrix}\right]X_j^T)}}. \quad (5)$$

In order to approximate the ML-estimation defined in (4), we calculate (5) for an approximate ML-labeling: we define a generalized ML-approximation of $\mathbb{P}[\mathbf{X}_i = X_i|\mathbf{M} = M, \mathbf{X}_L = X_L]$ by generalizing (5) to condition directly on the set of SDP-output vector-labels $X'_U \in \mathbb{R}^{(n-k)\times(n-k+r)}$. Thus, the new MLE-approximation for $\mathbb{P}[\mathbf{X}_i = X_i|\mathbf{M} = M, \mathbf{X}_L = X_L]$ becomes

$$\hat{\mathbb{P}}[\mathbf{X}_i = X_i|\mathbf{M} = M, \mathbf{X}_L = X_L, \mathbf{X}_{U_{\neg i}} = X'_{U_{\neg i}}] = \frac{e^{\frac{r-1}{r}(M_i\left[\begin{smallmatrix} X'_{U_{\neg i}} \\ X_L \end{smallmatrix}\right]X_i^T)}}{\sum_{X_j \in \triangle_r} e^{\frac{r-1}{r}(M_i\left[\begin{smallmatrix} X'_{U_{\neg i}} \\ X_L \end{smallmatrix}\right]X_j^T)}}, \quad (6)$$

where X'_U is the output of SDP(M, X_L). Note that $\hat{\mathbb{P}}$ still defines a probability distribution but it is no longer conditioned on a discrete community labeling.

4.2 Model Change

We use the ML-approximate distribution from (6) to define the 'model change' querying strategy. First, we define the model to be the $(n \times r)$-matrix:

$$\Phi(M, X_L, \triangle_r)_{i,j} \doteq \hat{\mathbb{P}}[\mathbf{X}_i = X_i^{(j)}|\mathbf{M} = M, \mathbf{X}_L = X_L, \mathbf{X}_{U_{\neg i}} = X'_{U_{\neg i}}] \quad (7)$$

where, $X'_U = \text{SDP}(M, X_L)$ and $X_i^{(j)}$ is the j^{th} vector-label in $\triangle_r$. For any already labeled node i, the probability distribution $\hat{\mathbb{P}}[\mathbf{X}_{L\text{-}i} = X_{L\text{-}i}^{(j)}|\mathbf{M} = M, \mathbf{X}_L = X_L, \mathbf{X}_{U_{\neg i}} = X'_{U_{\neg i}}]$ is defined to be the delta-function with mass 1 on the node's true label. We define the model change, δ, to be the sum of total variation distances between each node's probability distribution before and after a particular assignment $\mathbf{X}_q = X_q$:

$$\delta(\Phi, X_q) \doteq \|\Phi(M, [X_L, X_q], \triangle_r) - \Phi(M, X_L, \triangle_r)\|_T. \quad (8)$$

Thus, in order to maximize the Expected Model Change (EMC):

$$\text{EMC}(M, X'_U, X_L, \triangle_r) \doteq \sum_{X_q \in \triangle_r} \hat{\mathbb{P}}[\mathbf{X}_q = X_q|\mathbf{M} = M, \mathbf{X}_L = X_L, \mathbf{X}_{U_{\neg q}} = X'_{U_{\neg q}}] \cdot \delta(\Phi, X_q) \quad (9)$$

we query the label of node

$$q = \text{MEMC}(M, X'_U, X_L, \triangle_r) = \underset{q \in U}{\text{argmax}}\ EMC(M, X'_U, X_L, \triangle_r), \quad (10)$$

4.3 The algorithm

Pseudo-code for the entire active MEMC algorithm is presented in Fig. 2. The procedure

MEMC-Active(M, X_L, r, Q)
Input: M, X_L, r, Q: query budget
Output: $X \in \triangle_r^n$
1. For *queried* = 1 to Q
   a) $X' = \text{SDP}(M, X_L)$
   b) If $|\text{unique}(X_L)| < r$
      i) $\triangle_r = \text{bestFitSimplex}(X')$
      ii) $q = \text{Anchor}(M, X'_U, X_L, \triangle_r)$
   Else
      i) $\triangle_r = \text{unique}(X_L)$
      ii) $q = \text{MEMC}(M, X'_U, X_L, \triangle_r)$
   c) $X_q = \text{Label}(\mathbf{X}_q)$
   d) $X_L = \left[\begin{smallmatrix} X_L \\ X_q \end{smallmatrix}\right]$
2. $X = \text{Semi-Supervised}(M, X_L, r)$

Figure 2: The *Active Learning* MEMC algorithm

Anchor$(M, X'_U, X_L, \triangle_r)$ is a querying mechanisms for the early stages when X_L may not yet contain all existing community labels. We further elaborate on the anchor nodes selection and additional speedup steps for the SDP in the supplementary material.

5 Sample Complexity Analysis

In this section we provide theoretical sample complexity analysis to the MEMC criterion. As will be shown below, and in our empirical validation in Fig. 3, the error reduction rates that MEMC exhibits for the SBM differ throughout the active learning process and are characterized by 2 to 3 different phases, depending on the error type and the SNR regime. Therefore, our sample complexity analysis provides a more informative picture by proving the number of queries needed to recover the communities by MEMC, instead of the arbitrary error complexity analysis that is typical for active learning in standard classification tasks.

To facilitate our analysis we consider here the 2-community symmetric - $SSBM(n, 2, a, b)$ where $p = \frac{a}{n}$ and $q = \frac{b}{n}$. Our results do not loose their utility for $r > 2$ communities. We consider in the analysis the two different settings of SBM graphs: above and below the SNR detection threshold (see definition in Theorem 1).

To set the stage for our analysis we first define the differential degree of a node with respect to an assignment $\tilde{X}$ of labels:

Definition 5. *The Differential degree of a node v_i w.r.t to labeling $\tilde{X}$ is defined as*

$$d_{\tilde{X}}(v_i) = |\{v_j \in V | v_j \sim v_i, \tilde{X}_i = \tilde{X}_j\}| - |\{v_j \in V | v_j \sim v_i, \tilde{X}_i \neq \tilde{X}_j\}| \quad (11)$$

A node v has a *majority* with respect to a labeling $\tilde{X}$ if $d_{\tilde{X}}(v) > 0$. Otherwise, v has a *minority*. We characterize two types of possible errors in the labeling approximation $\tilde{X}$ with respect to the true labels X:

Definition 6. *Type-1 error: $\tilde{X}_i \neq X_i$ where v_i has a majority with respect to X.*

Definition 7. *Type-2 error: $\tilde{X}_i \neq X_i$ where v_i has a minority with respect to X.*

Clearly, type 1 and type 2 error nodes comprise the total of possible errors. As will be shown below, the sample complexity of each error type is qualitatively different and depends on the stage of the active learning process, or, more specifically, on the distribution of the currently unlabeled nodes differential degree.

We note that in the problem setting of SBM the Maximum-a-Posteriori (MAP) estimator is such that it assigns the label

$$\tilde{X}_i = \arg\max_{\tilde{X}_i} P\{X_i = \tilde{X}_i | S = s, R = s\}, \quad (12)$$

where S and R are the random variables representing the number of neighbors of v_i that have similar and opposite labels, respectively. Because label assignments

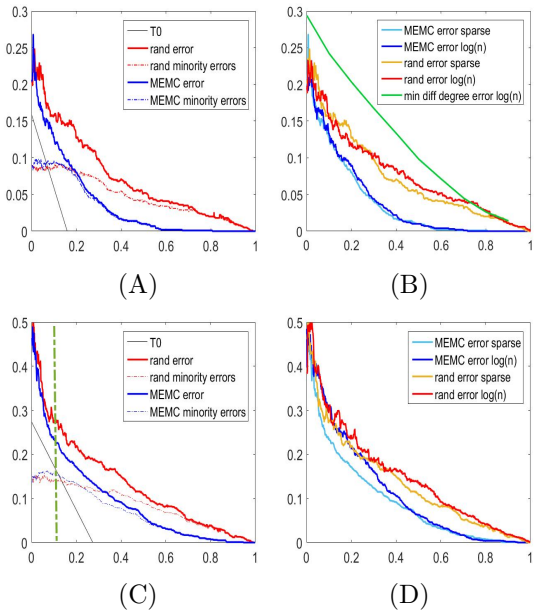


Figure 3: Comparison of MEMC and random error vs. fraction of data queried. **A,B:** $SNR \approx 1.3$, sparse (A) logarithmic (B). **C,D:** $SNR \approx 0.6$ sparse (C) logarithmic (D). Dashed vertical line indicating transition from super-linear to linear phase.

are an independent process in SBM, the MAP estimator is equal to the ML estimator:

$$\tilde{X}_i = \arg\max_{\tilde{X}_i} P\{S = s, R = s | X_i = \tilde{X}_i\}. \quad (13)$$

Therefore, in order to facilitate our analysis we will consider the local MAP classier (shown above to be equivalent to the ML classier), and scalar labels.

5.1 The SNR > 1 case

For the case of SNR> 1 we provide our main result in Theorem 3 for recovery. Theorem 3 is also validated by our empirical observations in Fig. 3-A,B in section 6. In particular, our experiments in Fig. 3-A,B validate that there are 2 different phases in which different types of errors are corrected at different rates throughout the recovery via the active learning process. This observation encourages us to examine the sample complexity of a full recovery to provide a comprehensive analysis.

Theorem 3. *(Sample Complexity) Consider the network $M \sim SSBM(n, 2, p, q)$ such that $SNR > 1$. Let $\tilde{X} = SDP(M, X_L)$ be the approximate scalar labels of the nodes, and X_L is a set of known labels for a subset of nodes. Let the total of number of errors in $\tilde{X}$ be $m = m_1 + m_2$, where m_i corresponds to type-i error. Then the sample complexity of MEMC for full recovery*

is at most

$$m_1+(n-m_1)\Big(\exp\Big(-\Big(\sqrt{\frac{b}{2}}-\sqrt{\frac{a}{2}}\Big)^2\Big)+\sum_{k=1}^{-l_c}P(k;a,b)\Big), \tag{14}$$

where l_c is the critical differential degree such that a.a.s no nodes exists s.t. $d_X(v) < l_c$, and $P(k;a,b)$ is the Skellam distribution representing the probability of a difference between two Poisson random variables with means a and b. Specifically, $l_c = inf\{k|(P(k;a,b) \leq k) = o(n^{-1})\}$. On the other hand Random will require order n queries.

Proof sketch: The proof relies on three important auxiliary results - Lemmas 2, 3, and corollary 2 (the full proof of Theorem 3 and the auxiliary results are provided in the supplementary material). First, in Lemma 2 we prove that type-1 errors have higher model change than other nodes of similar degree. Therefore, MEMC will initially query the type-1 error nodes, corresponding to the first m_1 component in (14). Second, following the correction of type-1 error nodes MEMC triggers a search around zero differential degree for type-2 nodes which show preference for correcting type-2 errors over querying correctly assigned nodes, as per Corollary 2. Lastly, the type-2 error search range is proved to be symmetrically bounded around zero for a differential degree range that can be directly inferred from a and b's values as provided in Lemma 3. The random criterion will select all types of nodes at equal probability within an unbounded range of the possible differential degrees.

The sample complexity of the different algorithms for SNR>1 is summarized in Fig. 4, where we also plot the curve for a hypothetic optimal active learner which makes only T0 errors of type-2 and queries only them. We provide below the auxiliary results for prooving Theorem 3.

Lemma 2. *Consider the network $M \sim SSBM(n,2,p,q)$. Let $\tilde{X} = SDP(M, X_L)$ be the scalar labels of the nodes, and X_L is a set of known labels for a subset of nodes. Let v^1 be a node of differential degree $\delta > 0$ for which $\tilde{X}$ has made a type-1 error, let v^2 be a node of differential degree $-\delta$ for which $\tilde{X}$ has made a type-2 error, and let v^3 be a correctly assigned node with differential degree δ. Then EMC satisfies*

$$EMC(v^1) > EMC(v^2) = EMC(v^3). \tag{15}$$

Next, we prove the conditions under which MEMC will have preference to correct type-2 nodes over querying correctly assigned nodes. In particular, Corollary 2 guarantees that if there are type-2 error nodes with absolute differential degree that is lower than other

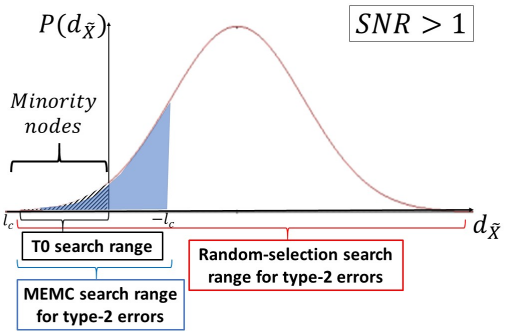


Figure 4: Sample complexity for recovery in the $SNR > 1$ regime. The probability mass of the differential degree $d_{\tilde{X}}$ is plotted, with regions marked for minority nodes and MEMC search region within critical degrees $[l_c, -l_c]$.

node's differential degree (as typical for type-2 nodes), they will be queried by MEMC:

Corollary 2. *Given $M \sim SSBM(n,2,a,b)$. Let v_j^2 be a type-2 node, and let v_i^3 be a correctly assigned node. If $|d_{\tilde{X}}(v_j^2)| < |d_{\tilde{X}}(v_i^3)|$ then $EMC(v_j^2) > EMC(v_i^3)$.*

Lastly, Lemma 3 provides bounds on the probability of having minority nodes which cause type-2 errors. While majority nodes can be classified correctly by MAP and ML classifiers once most of their neighbors are correctly identified, minority error nodes (i.e. type-2 error nodes) need to be queried directly in order to be corrected. Therefore estimating their proportion in the overall set is crucial for bounding sample complexity analysis for querying type-2 error nodes. In particular, Lemma 3 provides a bound on the probability of a minority node that can be used to derive a bound on the minimal differential degree l_c that minority nodes could have.

Lemma 3. *Given $SSBM(n,2,a,b)$, the probability of a minority node $P(c_{out} \geq c_{in})$ satisfies*

$$\frac{\exp(-(\sqrt{\frac{b}{2}}-\sqrt{\frac{a}{2}})^2)}{(\frac{a+b}{2})^2}-\frac{\exp(-(\frac{b}{2}+\frac{a}{2}))}{\sqrt{2}ab}-\frac{\exp(-(\frac{b}{2}+\frac{a}{2}))}{2ab}$$
$$\leq P(c_{out} \geq c_{in}) \leq \exp\Big(-\Big(\sqrt{\frac{b}{2}}-\sqrt{\frac{a}{2}}\Big)^2\Big). \tag{16}$$

5.2 The SNR < 1 case

Below the SNR threshold there is no unsupervised consistent estimator that can provide clustering better than a trivial guess [Mossel(2015)]. In the semi-supervised setting, we observe a super-linear reduction in error with each query at the initial stage of querying

(see Fig. 3 C and D). This fast improvement rate is due to the concentration of the differential degree (for $\tilde{X}$) at zero, causing every query-flip to trigger cascades of label flips. We first define the phenomena of *cascades*, and quantify its rate in Lemma 4 below.

Definition 8. *(Cascades) Consider the network $M \sim SSBM(n, 2, p, q)$. Let $\tilde{X} = Alg_\Psi(M, X_L)$ be the predicted labels of the nodes according to a recovery algorithm Ψ, and X_L is a set of known labels for a subset of nodes. Let v_q be a node whose label has been queried. Then a cascade occurs if there exists at least one path $v_{q_0} \sim v_{q_1}, ..., v_{q_{l-1}} \sim v_{q_l}$, where $v_{q_0} = v_q$, $v_{q_i} \sim v_{q_{i+1}}$, and $l \geq 2$, such that $\tilde{x}_{q_i} \neq \tilde{x}^{up}_{q_i}$ for every $0 \leq i \leq l$, where $\tilde{X}^{up} = Alg_\Psi(M, X_L \cup x_q)$.*

Cascades give rise to the super-linear error reduction observed in the $SNR < 1$ regime, where the expected differential degree with respect to $\tilde{X}$ is concentrated at and closely around 0. Lemma 4 provides the estimation of the cascades effect on the overall community detection accuracy as a function of a and b:

Lemma 4. *Consider the connected network $M \sim SSBM(n, 2, p, q)$ with average degree d, and $SNR < 1$, such that in expectation the differential degree $d_{\tilde{X}}(v)$ with respect to the $\tilde{X}$ approximation is zero. Then the expected number of majority(minority) nodes that correctly change their assignment after a query v_q followed by a MAP update is at least*

$$N_{maj(min)} = \frac{dp_{maj(min)}}{1 - dp_{maj(min)}}(1 - (dp_{maj(min)})^{\log_d(n)}), \tag{17}$$

where $p_{maj(min)}$ is the probability of a majority(minority) node flipping its label correctly following its neighbor change of label and a MAP update. $p_{maj} = (1 - 2\bar{p}_{min}) \cdot \left(\frac{a-b}{4(a+b)}\right)$, where $\bar{p}_{min}$ is assigned with the upper bound for the probability of a minority node in Equation (16), and, $p_{min} = \bar{p}_{min} \cdot \left(\frac{b-a}{4(a+b)}\right)$.

We note the dependency of (17) in n in the sparse regime: if $dp_{maj(min)} \approx d$, then$(1 - (dp_{maj(min)})^{\log_d(n)} \approx O(n)$ which will result in an order n correction!

Following the super linear error reduction phase, MEMC exhibits a linear reduction with preference to type-1 nodes succeeded by a bounded search for type-2 nodes, similarly to what was shown for the SNR> 1 case. We note that these 3 stages of active learning are also observed in Fig. 3-C,D. We provide our main results on the sample complexity for recovery in the SNR < 1 case in the following Theorem:

Theorem 4. *(Sample Complexity) Given the network $M \sim SSBM(n, 2, p, q)$ with average degree d, and $SNR < 1$. Let $m = m_1 + m_2$ be the total number of errors in $\tilde{X}$, where m_i corresponds to errors of type-i. Then the expected sample complexity of MEMC is*

$$\begin{aligned} &\frac{n}{2(N_{maj} + N_{min})} + \Big(m_1 - \frac{nN_{maj}}{2(N_{maj} + N_{min})}\Big) \\ &+ \Big(n - \frac{nN_{min}}{2(N_{maj} + N_{min})} - m_1\Big) \\ &\cdot \Big(\sum_{k=1}^{-l_c} P(k; a, b) + \exp\Big(-\Big(\sqrt{\frac{b}{2}} - \sqrt{\frac{a}{2}}\Big)^2\Big)\Big), \end{aligned} \tag{18}$$

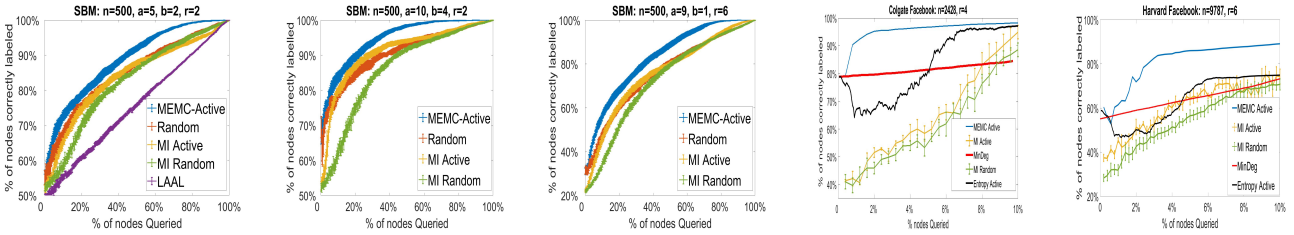
where $N_{maj(min)}$ are as defined in Lemma 4. Conversely, the sample complexity of Random is n.

Theorem 4 concludes that MEMC number of queries is significantly lower than the Random criterion, even though both MEMC and Random demonstrate initially a super-linear error reduction. To this end, there are two important observations that address MEMC's superiorly: First, following Theorem 4, type-1 nodes are recovered initially and the search for type-2 nodes is within a bounded differential degree range $[-l_c, l_c]$, where they are concentrated. Thus, MEMC search covers only a fraction of the n nodes, while it still guarantees recovery of the type-2 errors. On the other hand, Random slection samples the full range of differential degrees for both types of error nodes. Second, in the sparse regime multiple connected components may exist. MEMC has preference to select the components of largest diameter, which will introduce higher model change and therefore deeper correction cascades, as seen in Fig. 3 C,D. On the other hand, Random does not inflict preference for larger connected components.

5.3 Logarithmic vs. Sparse Degree Regimes

In the $SNR > 1$ regime both sparse and logarithmic degree regimes have similar behavior (see Fig. 3 A and B). This similarity is a direct result of the general preference of MEMC to correct erroneous nodes. Moreover, since cascades do not take place in the $SNR > 1$ regime, the degree regime, which influences the diameter of connected components, plays no role here.

In the $SNR < 1$ case in the sparse regime the probability of zero differential degree is significantly higher than that of the logarithmic regime where the differential degree distribution is much more spread-out due to the higher degree magnitude. In the sparse degree regime significantly more mass of the distribution is localized at zero because a and b are small constants. As a result, correction cascades will occur with higher probability in the sparse regime.



(A) (B) (C) (D) (E)

Figure 5: Accuracy vs. Percentage of nodes queried. SBM: (A) $a = 5$, $b = 2$, $r = 2$, (B) $a = 10$ $b = 4$ $r = 2$, (C) $a = 9$, $b = 1$, $r = 6$. Social (D) Colgate FB, $r = 4$, (E) Harvard FB, $r = 6$.

6 Experimental Results

Empirical sample-complexity study. To validate our theoretical analysis we provide an empirical study of the optimality of MEMC. The study comprises of an error analysis including type-1,2 error evolution. MEMC's curves are also compared with those of a Random selection. We also provide the curve for a hypothetic optimal active algorithm which makes only T0 errors of type-2 and queries only them. The study covers SNR that is above (Fig. 3 A, B) and below (Fig. 3 C, D) the detection threshold, and the cases of sparse vs. logarithmic average degree regimes. We underline in the following the main observations:

1. $SNR > 1$: Type-2 errors remain fixed as type-1 errors are corrected, confirming Lemma 2. Random selection, conversely, presents a slower error improvement, due to it's unbounded search. Note T0 is larger than actual type-2 errors since some minority nodes are correctly classified by chance.

2. $SNR > 1$: MEMC error curve aligns early with the type-2 error curve indicating transition to querying type-2 nodes, confirming Corollary 2.

3. $SNR > 1$: The error curve for the minimal differential degree sampling methodology of [Gadde et al.(2016)] (designed for SNR> 1 at logarithmic regime) is presented in Fig. 3-B, demonstrating that using minimal differential degree without updates is sub-optimal.

4. $SNR < 1$: A *super-linear* error reduction is observed for both MEMC and Random. However, MEMC consistently has a significant advantage (of about 7%) over Random due to selecting the cascades introducing a larger model change.

5. $SNR < 1$: the sparse regime has deeper and longer cascades phase than the logarithmic regime, in line with the sparse SBM having more mass localized at zero differential degree.

6. $SNR < 1$: The super-linear phase is followed by a linear phase as in the SNR> 1 regime. At 50% queries MEMC's error curve aligns with the type-2 error curve. Perfect recovery is achieved at 80% for MEMC vs. 100% for Random.

Performance experiments. We present experiments for 2- and multi-community detection on SBM networks and real-world social networks (see data links and details at [Leskovec and Krevl(2014)]). The authors suggest that the student's class years form interesting ground truth communities in Facebook social network. We test the ability of algorithms 'Active,' and various active learners to cluster Colgate University students according to the 4 class years 2006-09 and Harvard University students according to the 6 class years 2004-09.

The performance of MEMC ('MEMC-Active') is compared with the two known active learners developed for SBMs [Moore et al.(2011)] ('MI') and [Gadde et al.(2016)] ('MinDeg'), and with the graph-based active learning algorithm of [Kushnir(2014)] ('LAAL'). A random selection ('Random') and an Entropy ('Entropy-Active) criterion using our SDP are also presented as baselines to MEMC's criterion. The advantage of MEMC is prominent: 'MI' uses max mutual information of a query with the current labeling and therefore has sub-optimal performance when labelling is still scarce. MinDeg uses the minimal differential degree without updates (applicable only for SNR> 1). 'Entropy' uses our SDP formulation, however, the max-entropy criterion focuses on uncertainty and does not explore nodes that can have more extensive model change. Thus only towards later stages of refinement the entropy provides improved gains. Finally, 'LAAL' assumes smoothness of the labeling function over the graph which does not exist for SBMs.

References

A. E. Allahverdyan, G. Ver Steeg, and A. Galstyan. Community detection with and without prior in-

formation. *EPL (Europhysics Letters)*, 90(1):18002, 2010.

A. S. Bandeira, N. Boumal, and V. Voroninski. On the low-rank approach for semidefinite programs arising in synchronization and community detection. In Vitaly Feldman, Alexander Rakhlin, and Ohad Shamir, editors, *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 361–382, Columbia University, New York, New York, USA, 23–26 Jun 2016. PMLR.

J. Cheng, M. Leng, L. Li, H. Zhou, and X. Chen. Active semi-supervised community detection based on must-link and cannot-link constraints. *PLoS One*, 9(10):18002, 2014.

A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E : Statistical, Nonlinear, and Soft Matter Physics*, 84:066106, 2011.

E. Eaton and R. Mansbach. A spin-glass model for semi-supervised community detection. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, pages 900–906. AAAI Press, 2012.

S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75 – 174, 2010.

A. Freytag, E. Rodner, and J. Denzler. Selecting influential examples: Active learning with expected model output changes. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 562–577, Cham, 2014. Springer International Publishing.

A. Gadde, E. En Gad, S. Avestimehr, and A. Ortega. Active learning for community detection in stochastic block models. *CoRR*, 1605 02372, 2016.

M. X. Goemans and D.P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42:1115–1145, 1995.

P. Holland, K. Laskey, and S. Leinhardt. Stochastic blockmodels: First steps. *Social Networks - SOC NETWORKS*, 5:109–137, 06 1983.

E. Mossel and J. Xu. Local algorithms for block models with side information *7th Innovations in Theoretical Computer Science (ITCS)*, Jan. 2016.

H. Saad, and A. Nosratinia. Community Detection with Side Information: Exact Recovery under the Stochastic Block Model. *IEEE Journal of Selected Topics in Signal Processing*, May. 2018

V. Kanade, E. Mossel, and T. Schramm. Global and local information in clustering labeled block models. *IEEE Trans. Information Theory*, 62(10):5906–5917, 2016.

Erdős,Rényi. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, 5:17-61, 1960.

D. Kushnir. Active-transductive learning with label-adapted kernels. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 462–471, New York, NY, USA, 2014. ACM.

M. Leng, Y. Yao, J. Cheng, W. Lv, and X. Chen. Active semi-supervised community detection algorithm with label propagation. In Weiyi Meng, Ling Feng, Stéphane Bressan, Werner Winiwarter, and Wei Song, editors, *Database Systems for Advanced Applications*, pages 324–338, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

S. Liu. Maximum likelihood estimation and inference: With examples in r, sas, and admb by russell b. millar. *International Statistical Review*, 80(2):346–346, 2012.

L. Massoulié. Community detection thresholds and the weak ramanujan property. In *Proceedings of the Forty-sixth Annual ACM Symposium on Theory of Computing*, STOC '14, pages 694–703, New York, NY, USA, 2014. ACM.

C. Moore, X. Yan, Y. Zhu, J.-Baptiste Rouquier, and Terran Lane. Active learning for node classification in assortative and disassortative networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 841–849, New York, NY, USA, 2011. ACM.

E. Mossel, J. Neeman, and A. Sly. Consistency thresholds for the planted bisection model. In *Proceedings of the Forty-seventh Annual ACM Symposium on Theory of Computing*, STOC '15, pages 69–75, New York, NY, USA, 2015. ACM.

E. Mossel, J. Neeman, and A. Sly. Reconstruction and estimation in the planted partition model. *Probability Theory and Related Fields*, 162(3):431–461, Aug 2015.

R. R. Nadakuditi, and M. EJ Newman. Graph spectral and detectability of community structure in networks. Physical Review Letters 108(18): 188701, 2012.

S.-Y. Yun and A. Proutiere. Optimal Cluster Recovery in the Labeled Stochastic Block Model. ArXiv e-prints, October 2015.

E. Abbe. Community detection and stochastic block models: recent developments *Journal of Machine Learning Research* 18, pages 1–177, 2017

B. Settles. Active learning literature survey. Technical report, 2010.

L. Tang and H. Liu. *Community Detection and Mining in Social Media*, volume 2. 01 2010.

A. L. Traud, P. J. Mucha, and M. A. Porter. Social structure of facebook networks. *Physica A: Statistical Mechanics and its Applications*, 391(16):4165–4180, 2012.

A. Vezhnevets, J. Buhmann, and V. Ferrari. Active learning for semantic segmentation with expected change. 06 2012.

L. Yang, D. Jin, X. Wang, and X. Cao. Active link selection for efficient semi-supervised community detection. *Scientific Reports*, 5, 2015.

P. Zhang, C. Moore, and L. Zdeborova. Phase transitions in semisupervised clustering of sparse networks. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 90, 04 2014.

J. Leskovec, and A. Krevl. Stanford Large Network Dataset Collection. *http://snap.stanford.edu/data*. 06 2014.

G. M. Kamath and E. Sasoglu and D. N. C. Tse. Optimal Haplotype Assembly from High-Throughput Mate-Pair Reads. In *http://arxiv.org/abs/1502.01975*, 2015.