The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

The Geometric Block Model

Sainyam Galhotra, Arya Mazumdar, Soumyabrata Pal, Barna Saha
College of Information and Computer Sciences
University of Massachusetts Amherst
Amherst, MA 01003
{sainyam,arya,spal,barna}@cs.umass.edu

Abstract

To capture the inherent geometric features of many community detection problems, we propose to use a new random graph model of communities that we call a *Geometric Block Model*. The geometric block model generalizes the random geometric graphs in the same way that the well-studied stochastic block model generalizes the Erdös-Renyi random graphs. It is also a natural extension of random community models inspired by the recent theoretical and practical advancement in community detection. While being a topic of fundamental theoretical interest, our main contribution is to show that many practical community structures are better explained by the geometric block model. We also show that a simple triangle-counting algorithm to detect communities in the geometric block model is near-optimal. Indeed, even in the regime where the average degree of the graph grows only logarithmically with the number of vertices (sparse-graph), we show that this algorithm performs extremely well, both theoretically and practically. In contrast, the triangle-counting algorithm is far from being optimum for the stochastic block model. We simulate our results on both real and synthetic datasets to show superior performance of both the new model as well as our algorithm.

1 Introduction

The *planted-partition* model or the *stochastic block model* (SBM) is a random graph model for community detection that generalizes the well-known Erdös-Renyi graphs (Holland, Laskey, and Leinhardt 1983; Dyer and Frieze 1989; Decelle et al. 2011; Abbe and Sandon 2015a; Abbe, Bandeira, and Hall 2016; Hajek, Wu, and Xu 2015; Chin, Rao, and Vu 2015; Mossel, Neeman, and Sly 2015). Consider a graph $G(V, E)$, where $V = V_1 \sqcup V_2 \sqcup \cdots \sqcup V_k$ is a disjoint union of k clusters denoted by $V_1, \ldots, V_k$. The edges of the graph are drawn randomly: there is an edge between $u \in V_i$ and $v \in V_j$ with probability $q_{i,j}, 1 \leq i, j \leq k$. Given the adjacency matrix of such a graph, the task is to find exactly (or approximately) the partition $V_1 \sqcup V_2 \sqcup \cdots \sqcup V_k$ of V.

This model has been incredibly popular both in theoretical and practical domains of community detection, and the aforementioned references are just a small sample. Recent theoretical works focus on characterizing sharp threshold of recovering the partition in the SBM. For example, when there are only two communities of exactly equal sizes, and the inter-cluster edge probability is $\frac{b \log n}{n}$ and intra-cluster edge probability is $\frac{a \log n}{n}$, it is known that perfect recovery is possible if and only if $\sqrt{a} - \sqrt{b} > \sqrt{2}$ (Abbe, Bandeira, and Hall 2016; Mossel, Neeman, and Sly 2015). The regime of the probabilities being $\Theta\Big(\frac{\log n}{n}\Big)$ has been put forward as one of most interesting ones, because in an Erdös-Renyi random graph, this is the threshold for graph connectivity (Bollobás 1998). This result has been subsequently generalized for k communities (Abbe and Sandon 2015a; 2015b; Hajek, Wu, and Xu 2016) (for constant k or when $k = o(\log n)$), and under the assumption that the communities are generated according to a probabilistic generative model (there is a prior probability p_i of an element being in the ith community) (Abbe and Sandon 2015a). Note that, the results are not only of theoretical interest, many real-world networks exhibit a "sparsely connected" community feature (Leskovec et al. 2008), and any efficient recovery algorithm for SBM has many potential applications.

One aspect that the SBM does not account for is a "transitivity rule" ('friends having common friends') inherent to many social and other community structures. To be precise, consider any three vertices x, y and z. If x and y are connected by an edge (or they are in the same community), and y and z are connected by an edge (or they are in the same community), then it is more likely than not that x and z are connected by an edge. This phenomenon can be seen in many network structures - predominantly in social networks, blog-networks and advertising. SBM, primarily a generalization of Erdös-Renyi random graph, does not take into account this characteristic, and in particular, probability of an edge between x and z there is independent of the fact that there exist edges between x and y and y and z. However, one needs to be careful such that by allowing such "transitivity", the simplicity and elegance of the SBM is not lost.

Inspired by the above question, we propose a random graph community detection model analogous to the stochastic block model, that we call the *geometric block model* (GBM). The GBM depends on the basic definition of the *random geometric graph* that has found a lot of practical use in wireless networking because of its inclusion of the notion of proximity between nodes (Penrose 2003).

Definition. A random geometric graph (RGG) on n vertices has parameters n, an integer $t > 1$ and a real number $\beta \in [-1, 1]$. It is defined by assigning a vector $Z_i \in \mathbb{R}^t$ to vertex $i, 1 \leq i, n$, where $Z_i, 1 \leq i \leq n$ are independent and identical random vectors uniformly distributed in the Euclidean sphere $\mathcal{S}^{t-1} \equiv \{x \in \mathbb{R}^t : \|x\|_{\ell_2} = 1\}$. There will be an edge between vertices i and j if and only if $\langle Z_i, Z_j \rangle \geq \beta$.

Note that, the definition can be further generalized by considering Z_is to have a sample space other than $\mathcal{S}^{t-1}$, and by using a different notion of distance than inner product (i.e., the Euclidean distance). We simply stated one of the many equivalent definitions (Bubeck et al. 2016).

Random geometric graphs are often proposed as an alternative to Erdös-Renyi random graphs. They are quite well studied theoretically (though not nearly as much as the Erdös-Renyi graphs), and very precise results exist regarding their connectivity, clique numbers and other structural properties (Gupta and Kumar 1998; Penrose 1991; Devroye et al. 2011; Avin and Ercal 2007; Goel, Rai, and Krishnamachari 2005). For a survey of early results on geometric graphs and the analogy to results in Erdös-Renyi graphs, we refer the reader to (Penrose 2003). A very interesting question of distinguishing an Erdös-Renyi graph from a geometric random graph has also recently been studied (Bubeck et al. 2016). This will provide a way to test between the models which better fits a scenario, a potentially great practical use.

As mentioned earlier, the "transitivity" feature led to random geometric graphs being used extensively to model wireless networks (for example, see (Haenggi et al. 2009; Bettstetter 2002)). Surprisingly, however, to the best of our knowledge, random geometric graphs are never used to model community detection problems. In this paper we take the first step towards this direction. Our main contributions can be classified as follows.

- We define a random generative model to study canonical problems of community detection, called the *geometric block model* (GBM). This model takes into account a measure of proximity between nodes and this proximity measure characterizes the likelihood of two nodes being connected when they are in same or different communities. The geometric block model inherits the connectivity properties of the random geometric graphs, in particular the likelihood of "transitivity" in triplet of nodes (or more).
- We experimentally validate the GBM on various real-world datasets. We show that many practical community structures exhibit properties of the GBM. We also compare these features with the corresponding notions in SBM to show how GBM better models data in many practical situations.
- We propose a simple motif-based efficient algorithm for community detection on the GBM. We rigorously show that this algorithm is optimal up to a constant fraction (to be properly defined later) even in the regime of sparse graphs (average degree $\sim \log n$).
- The motif-counting algorithms are extensively tested on both synthetic and real-world datasets. They exhibit very good performance in three real datasets, compared to the spectral-clustering algorithm (see Section 5). Since simple motif-counting is known to be far from optimum in stochastic block model (see Section 4), these experiments give further validation to GBM as a real-world model.

Given any simple random graph model, it is possible to generalize it to a random block model of communities much in line with the SBM. We however stress that the geometric block model is perhaps the simplest possible model of real-world communities that also captures the transitive/geometric features of communities. Moreover, the GBM explains behaviors of many real world networks as we will exemplify subsequently.

2 The Geometric Block Model and its Validation

Let $V \equiv V_1 \sqcup V_2 \sqcup \cdots \sqcup V_k$ be the set of vertices that is a disjoint union of k clusters, denoted by $V_1, \ldots, V_k$. Given an integer $t \geq 2$, for each vertex $u \in V$, define a random vector $Z_u \in \mathbb{R}^t$ that is uniformly distributed in $\mathcal{S}^{t-1} \subset \mathbb{R}^t$, the $t-1$-dimensional sphere.

Definition (Geometric Block Model ($V, t, \beta_{i,j}, 1 \leq i < j \leq k$)). Given V, t and a set of real numbers $\beta_{i,j} \in [-1, 1], 1 \leq i \leq j \leq k$, the geometric block model is a random graph with vertices V and an edge exists between $v \in V_i$ and $u \in V_j$ if and only if $\langle Z_u, Z_v \rangle \geq \beta_{i,j}$.

The case of $t = 2$: In this paper we particularly analyze our algorithm for $t = 2$. In this special case, the above definition is equivalent to choosing random variable θ_u uniformly distributed in $[0, 2\pi]$, for all $u \in V$. Then there will be an edge between two vertices $u \in V_i, v \in V_j$ if and only if $\cos\theta_u \cos\theta_v + \sin\theta_u \sin\theta_v = \cos(\theta_u - \theta_v) \geq \beta_{i,j}$ or $\min\{|\theta_u - \theta_v|, 2\pi - |\theta_u - \theta_v|\} \leq \arccos \beta_{i,j}$. This in turn, is equivalent to choosing a random variable X_u uniformly distributed in $[0, 1]$ for all $u \in V$, and there exists an edge between two vertices $u \in V_i, v \in V_j$ if and only if

$$\min\{|X_u - X_v|, 1 - |X_u - X_v|\} \leq r_{i,j},$$

where $r_{i,j} \in [0, \frac{1}{2}], 0 \leq i, j \leq k$, are a set of real numbers.

For the rest of this paper, we concentrate on the case when $r_{i,i} = r_s$ for all $i \in \{1, \ldots, k\}$, which we call the "intra-cluster distance" and $r_{i,j} = r_d$ for all $i, j \in \{1, \ldots, k\}, i \neq j$, which we call the "inter-cluster distance," mainly for the clarity of exposition. To allow for edge density to be higher inside the clusters than across the clusters, assume $r_s \geq r_d$.

The main problem that we seek to address is following. Given the adjacency matrix of a geometric block model with k clusters, and t, r_d, r_s, $r_s \geq r_d$, find the partition $V_1, V_2, \ldots, V_k$.

We next give two examples of real datasets that motivate the GBM. In particular, we experiment with two different types of real world datasets in order to verify our hypothesis about geometric block model and the role of distance in the formation of edges. The first one is a dataset with academic collaboration, and the second one is a product purchase metadata from Amazon.

2.1 Motivation of GBM: Academic Collaboration

We consider the collaboration network of academicians in Computer Science in 2016 (data obtained from `csrankings.org`). According to area of expertise of the

Area 1	Area 2	same	different
MOD	AI	10	2
ARCH	MOD	6	1
ROB	ARCH	3	0
MOD	ROB	4	0
ML	MOD	7	1

Area	same	different
MOD	19	35
ARCH	13	15
ROB	24	16
AI	39	32
ML	14	42

Table 1: On the left we count the number of inter-cluster edges when authors shared same affiliation and different affiliations. On the right, we count the same for intra-cluster edges.

authors, we consider five different communities: Data Management (MOD), Machine Learning and Data Mining (ML), Artificial Intelligence (AI), Robotics (ROB), Architecture (ARCH). If two authors share the same affiliation, or shared affiliation in the past, we assume that they are geographically close. We would like to hypothesize that, two authors in the same communities might collaborate even when they are geographically far. However, two authors in different communities are more likely to collaborate only if they share the same affiliation (or are geographically close). Table 1 describes the number of edges across the communities. It is evident that the authors from same community are likely to collaborate irrespective of the affiliations and the authors of different communities collaborate much frequently when they share affiliations or are close geographically. This clearly indicates that the inter cluster edges are likely to form if the distance between the nodes is quite small, motivating the fact $r_d < r_s$ in the GBM.

2.2 Motivation of GBM: Amazon Metadata

The next dataset that we use in our experiments is the Amazon product metadata on SNAP (https://snap.stanford.edu/data/amazon-meta.html), that has 548552 products and each product is one of the following types {Books, Music CD's, DVD's, Videos}. Moreover, each product has a list of attributes, for example, a book may have attributes like ⟨"General", "Sermon", "Preaching"⟩. We consider the co-purchase network over these products. We make two observations here: (1) edges get formed (that is items are co-purchased) more frequently if they are similar, where we measure similarity by the number of common attributes between products, and (2) two products that share an edge have more common neighbors (no of items that are bought along with both those products) than two products with no edge in between.

Figures 1 and 2 show respectively average similarity of products that were bought together, and not bought together. From the distribution, it is quite evident that edges in a copurchase network gets formed according to distance, a salient feature of random geometric graphs, and the GBM.

We next take equal number of product pairs inside Book (also inside DVD, and across Book and DVD) that have an edge in-between and do not have an edge respectively. Figure 3 shows that the number of common neighbors when two products share an edge is much higher than when they do not–in fact, almost all product pairs that do not have an edge in between also do not share any common neighbor. This again strongly suggests towards GBM due to its transitivity property. On the other hand, this also suggests that SBM is not a good model for this network, as in SBM, two nodes having common neighbors is independent of whether they share an edge or not.

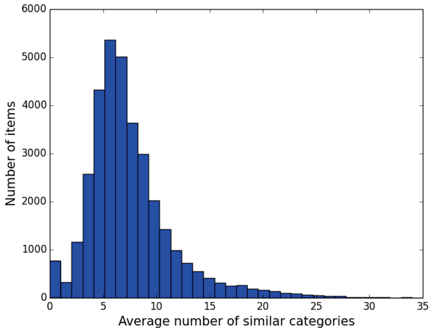


Figure 1: Histogram: similarity of products bought together (mean ≈ 6)

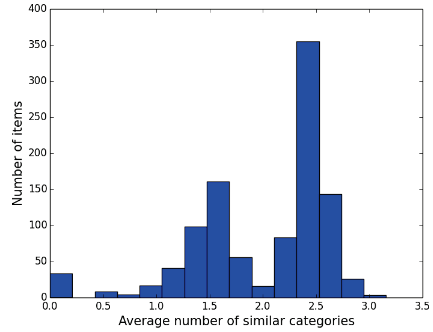


Figure 2: Histogram: similarity of products not bought together (mean≈ 2)

Difference between SBM and GBM. It is important to stress that the network structures generated by the SBM and the GBM are quite different, and it is significantly difficult to analyze any algorithm or lower bound on GBM compared to SBM. This difficulty stems from the highly correlated edge generation in GBM (while edges are independent in SBM). For this reason, analyses of the sphere-comparison algorithm and spectral methods for clustering on GBM cannot be derived as straight-forward adaptations. Whereas, even for simple algorithms, a property that can be immediately seen for SBM, will still require a proof for GBM.

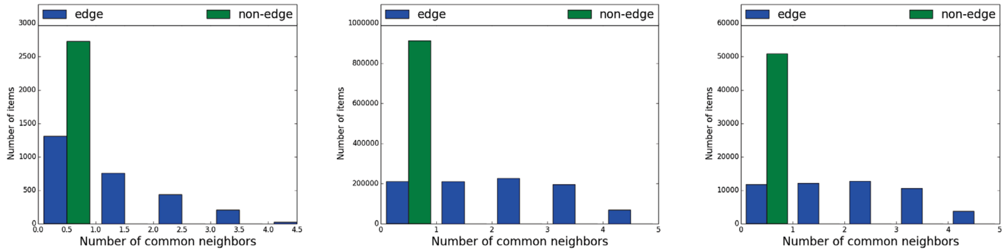


Figure 3: Histogram of common neighbors of edges and non-edges in the co-purchase network, from left to right: Book-DVD, Book-Book, DVD-DVD

3 The Motif-Counting Algorithm

Suppose, we are given a graph $G = (V, E)$ with k disjoint clusters, $V_1, V_2, ..., V_k \subseteq V$ generated according to $GBM(V, t, r_s, r_d, k)$. Our clustering algorithm is based on counting motifs, where a motif is simply defined as a configuration of triplets in the graph. Let us explain this principle by one particular motif, a triangle. For any two vertices u and v in V, where (u, v) is an edge, we count the total number of common neighbors of u and v. We show that this count is different when u and v belong to the same cluster, compared to when they belong to different clusters. We assume G is connected, because otherwise it is impossible to recover the clusters. For every pair of vertices in the graph that share an edge, we decide whether they are in the same cluster or not by this count of triangles. In reality, we do not have to check every such pair, instead we can stop when we form a spanning tree. At this point, we can transitively deduce the partition of nodes into clusters.

The main new idea of this algorithm is to use this triangle-count (or motif-count in general), since they carry significantly more information regarding the connectivity of the graph than an edge count. However, we can go to statistics of higher order (such as the two-hop common neighbors) at the expense of increased complexity. Surprisingly, the simple greedy algorithm that rely on triplets can separate clusters when r_d and r_s are $\Omega(\frac{\log n}{n})$, which is also a minimal requirement for connectivity of random geometric graphs (Penrose 2003). Therefore this algorithm is optimal up to a constant factor. It is interesting to note that this motif-counting algorithm is not optimal for SBM (as we observe), in particular, it will not detect the clusters in the sparse threshold region of $\frac{\log n}{n}$, however, it does so for GBM.

The pseudocode of the algorithm is described in Algorithm 1. The algorithm looks at individual pairs of vertices to decide whether they belong to the same cluster or not. We go over pair of vertices and label them same/different, till we have enough labels to partition the graphs into clusters.

At any stage, the algorithm picks up an unassigned node v and queries it with a node each from the already formed clusters. To decide whether a point belongs to a cluster, it calls a subroutine called `process`. The `process` function tries to infer if the node v belongs to the cluster V_i by first identifying a vertex $u \in V_i$ that has an edge with v, and then by counting the number of common neighbors of u and v to make a decision. This procedure is continued till all nodes in V are processed.

Algorithm 1: Graph recovery from GBM

```
Require: GBM $G = (V, E), r_s, r_d, k$
Ensure: $V = V_1 \sqcup \ldots \sqcup V_k$
$V_1, \ldots, V_k \leftarrow \emptyset$
for $v \in V$ do
    for $i \in \{1, 2, \ldots, k-1\}$ do
        if process($V_i, v, r_s, r_d$) then
            $V_i \leftarrow V_i \cup \{v\}$
            added← true
        end if
    end for
    if ¬ added then
        $V_k \leftarrow V_k \cup \{v\}$
    end if
end for
```

Algorithm 2: `process`

```
Require: $C, v, r_s, r_d$
Ensure: true/false
Choose $u \in C \mid (u, v) \in E$
count ← $|\{z : (z, u) \in E, (z, v) \in E\}|$
if $|\frac{\text{count}}{n} - E_S(r_d, r_s)| < |\frac{\text{count}}{n} - E_D(r_d, r_s)|$ then
    return true
end if
return false
```

The `process` function counts the number of common neighbors of two nodes and then compares the difference of the count with two functions of r_d and r_s, called E_D and E_S. Formulae for E_D and E_S are different when $r_s < 2r_d$ to $r_s \geq 2r_d$. We have compiled this in Table 2. In this table we have assumed that there are only two clusters of equal size. The functions change when the cluster sizes are different. Our analysis described in later sections can be used to calculate new function values. In the table, $u \sim v$ means u and v are in the same cluster.

Similarly, the `process` function can be run on other set of motifs (other patterns of triplets) by fixing two nodes. On considering a larger set of motifs, the `process` function can take a majority vote over the decisions received from different motifs. This is helpful to resolve the clusters even when the gap between r_s and r_d is small (by a constant factor than compared to just triangle motif).

Note that, our algorithm counts motifs only for edges, and does not count motifs for more than $n-1$ edges, as that many edges are sufficient to construct a spanning tree of the graph.

Motif	Distribution of count ($r_s > 2r_d$)		Distribution of count ($r_s \leq 2r_d$)	
	$u \sim v$	$u \nsim v$	$u \sim v$	$u \nsim v$
$z \mid (z,u) \in E, (z,v) \in E$	$\text{Bin}(\frac{n}{2} - 2, \frac{3r_s}{2}) + \text{Bin}(\frac{n}{2}, \frac{2r_d^2}{r_s})$; $E_S = \frac{3r_s}{4} + \frac{r_d^2}{r_s}$	$\text{Bin}(n - 2, 2r_d)$; $E_D = 2r_d$	$\text{Bin}(\frac{n}{2} - 2, \frac{3r_s}{2}) + \text{Bin}(\frac{n}{2}, 2r_d - \frac{r_s}{2})$; $E_S = \frac{r_s}{2} + r_d$	$\text{Bin}(n - 2, 2r_s - \frac{r_s^2}{2r_d})$ $E_D = 2r_s - \frac{r_s^2}{2r_d}$
$z \mid (z,u) \in E, (z,v) \notin E$	$\text{Bin}(\frac{n}{2} - 2, \frac{r_s}{2}) + \text{Bin}(\frac{n}{2}, \frac{2r_d(r_s-r_d)}{r_s})$; $E_S = \frac{r_s}{2} + \frac{r_d(r_s-r_d)}{r_s}$	$\text{Bin}(n-2, r_s - r_d)$; $E_D = r_s - r_d$	$\text{Bin}(n-2, \frac{r_s}{2})$; $E_S = \frac{r_s}{2}$	$\text{Bin}(n-2, \frac{r_s^2+2r_d^2-2r_sr_d}{r_d})$ $E_D = \frac{r_s^2+2r_d^2-2r_sr_d}{r_d}$
$z \mid (z,u) \notin E, (z,v) \in E$	$\text{Bin}(\frac{n}{2} - 2, \frac{r_s}{2}) + \text{Bin}(\frac{n}{2}, \frac{2r_d(r_s-r_d)}{r_s})$; $E_S = \frac{r_s}{2} + \frac{r_d(r_s-r_d)}{r_s}$	$\text{Bin}(n-2, r_s - r_d)$; $E_D = r_s - r_d$	$\text{Bin}(n-2, \frac{r_s}{2})$; $E_S = \frac{r_s}{2}$	$\text{Bin}(n-2, \frac{r_s^2+2r_d^2-2r_sr_d}{r_d})$ $E_D = \frac{r_s^2+2r_d^2-2r_sr_d}{r_d}$

Table 2: E_S, E_D values for different motifs considering different values of r_s and r_d, when there are two equal sized clusters. Here $\text{Bin}(n,p)$ denotes a binomial random variable with mean np.

4 Analysis of the Algorithm

The critical observation that we have to make to analyze the motif-counting algorithm is the fact that given a GBM graph $G(V,E)$ with two clusters $V = V_1 \sqcup V_2$, and a pair of vertices $u, v \in V : (u,v) \in E$, the events $\mathcal{E}_z^{u,v}, z \in V$ of any other vertex z being a common neighbor of both u and v are independent (this is obvious in SBM, but does not lead to the same result). However, the probabilities of $\mathcal{E}_z^{u,v}$ are different when u and v are in the same cluster and when they are in different clusters. Therefore the count of the common neighbors are going to be different, and substantially separated with high probability (due to being sums of independent random variables) for two vertices in cases when they are from the same cluster or from different clusters. This will lead the function `process` to correctly characterize two vertices as being from same or different clusters with high probability. Let us now show this more formally. We have the following two lemmas for a GBM graph $G(V,E)$ with two equal-sized (unknown) clusters $V = V_1 \sqcup V_2$, and parameters r_s, r_d.

Lemma 1. *For any two vertices $u, v \in V_i : (u,v) \in E, i = 1,2$ belonging to the same cluster, the count of common neighbors $C_{u,v} \equiv |\{z \in V : (z,u),(z,v) \in E\}|$ is a random variable distributed according to $\text{Bin}(\frac{n}{2} - 2, \frac{3r_s}{2}) + \text{Bin}(\frac{n}{2}, \frac{2r_d^2}{r_s})$ when $r_s > 2r_d$ and according to $\text{Bin}(\frac{n}{2} - 2, \frac{3r_s}{2}) + \text{Bin}(\frac{n}{2}, 2r_d - \frac{r_s}{2})$ when $r_s \leq 2r_d$, where $\text{Bin}(n,p)$ is a binomial random variable with mean np.*

Lemma 2. *For any two vertices $u \in V_1, v \in V_2 : (u,v) \in E$ belonging to different clusters, the count of common neighbors $C_{u,v} \equiv |\{z \in V : (z,u),(z,v) \in E\}|$ is a random variable distributed according to $\text{Bin}(n-2, 2r_d)$ when $r_s > 2r_d$ and according to $\text{Bin}(n-2, 2r_s - \frac{r_s^2}{2r_d})$ when $r_s \leq 2r_d$.*

Similar lemmas exists for other motifs as well (see the full version (Galhotra et al. 2017)). Here let us give the proof of Lemma 1. The proof of Lemma 2 will follow similarly. These expressions can also be generalized straightforwardly when the clusters are of unequal sizes, but we omit those for clarity of exposition.

Proof of Lemma 1. Let $X_w \in [0,1]$ be the uniform random variable associated with $w \in V$. Let us also denote by $d_{\text{L}}(X,Y) \equiv \min\{|X-Y|, 1-|X-Y|\}, X, Y \in \mathbb{R}$. Without loss of generality, assume $u, v \in V_1$. For any vertex $z \in V$, let $\mathcal{E}_z^{u,v} \equiv \{(u,z),(v,z) \in E\}$ be the event that z is a common neighbor. For $z \in V_1$,

$$\begin{aligned}
&\Pr(\mathcal{E}_z^{u,v}) = \Pr((z,u) \in E, (z,v) \in E \mid (u,v) \in E) \\
&= \frac{1}{r_s}\int_0^{r_s} \Pr((z,u) \in E, (z,v) \in E \mid d_{\text{L}}(X_u, X_v) = x)dx \\
&= \int_0^{r_s} \frac{1}{r_s}(2r_s - x)dx = \frac{3r_s}{2}.
\end{aligned}$$

For $z \in V_2$, assuming $\ell = \min(r_s, 2r_d)$, we have,

$$\begin{aligned}
&\Pr(\mathcal{E}_z^{u,v}) = \Pr((z,u) \in E, (z,v) \in E \mid (u,v) \in E) \\
&= \int_0^{\ell} \frac{1}{r_s} \Pr((z,u),(z,v) \in E \mid d_{\text{L}}(X_u, X_v) = x)dx \\
&= \int_0^{\ell} \frac{1}{r_s}(2r_d - x)dx = \begin{cases} \frac{2r_d^2}{r_s} & \text{if } 2r_d < r_s \\ 2r_d - \frac{r_s}{2} & \text{otherwise.} \end{cases}
\end{aligned}$$

Now since there are $\frac{n}{2} - 2$ points in $V_1 \setminus \{u,v\}$ and $\frac{n}{2}$ points in V_2, we have the statement of the lemma. □

The proof of Lemma 2 is similar and can be seen in the full version of this paper (Galhotra et al. 2017).

By leveraging the concentration of binomial random variables, in our algorithm we just check whether the count of common neighbors is closer to the average value of the random variable described in Lemma 1 or in Lemma 2. While more general statements are possible, we give a theorem concentrating on the special case when $r_s, r_d \sim \frac{\log n}{n}$.

Theorem 1. *If $r_s = \frac{a \log n}{n}$ and $r_d = \frac{b \log n}{n}$, $r_s > r_d$, Algorithm 1 can recover the clusters V_1, V_2 accurately with a probability of $1 - \frac{3}{n}$ if*

$$\begin{cases} \frac{(3a-2b)(a-2b)}{4a} \geq (\sqrt{\frac{3a}{4}} + \sqrt{\frac{b^2}{a}} + \sqrt{2b})\sqrt{6}, \textit{when } a \geq 2b \\ \frac{(a-b)(a-2b)}{2b} \geq (\sqrt{\frac{3a}{4}} + \sqrt{b - \frac{a}{4}} + \sqrt{2a - \frac{a^2}{2b}})\sqrt{6}, \textit{else} \end{cases}$$

.

Proof. Let us consider the case $r_s > 2r_d$ first. Let Z denote the random variable that equals the number of common neighbors of two nodes $u, v \in V : (u, v) \in E$. Let us also denote $\mu_s = \mathbb{E}(Z|u \sim v)$ and $\mu_d = \mathbb{E}(Z|u \nsim v)$, where $u \sim v$ means u and v are in the same cluster. We can easily find μ_s and μ_d from Lemmas 1, 2. We see that,

$$\mu_s - \mu_d = \frac{(n - O(1))(3r_s - 2r_d)(r_s - 2r_d)}{4r_s}$$
$$= \frac{(3a - 2b)(a - 2b)\log n}{4a} - O\Big(\frac{\log n}{n}\Big).$$

Now, since Z is a sum of independent binary random variables, using the Chernoff bound, $\Pr(Z < (1+\delta)\mathbb{E}(Z)) \leq \Pr(Z > (1+\delta)\mathbb{E}(Z)) \leq e^{-\delta^2\mathbb{E}(Z)/3} = \frac{1}{n^2}$, when $\delta = \sqrt{\frac{6\log n}{\mathbb{E}(Z)}}$. Now with probability at least $1 - \frac{3}{n^2}$ (since there are three binomial terms involved and they can have deviations more than $\sqrt{9a/2}\log n$, $\sqrt{6b^2/a}\log n$, and $\sqrt{12b}\log n$ with probability $\frac{1}{n^2}$ each), the algorithm will be successful to label correctly as long as, $\frac{(3a-2b)(a-2b)\log n}{4a} \geq \Big(\sqrt{\frac{3a}{4}} + \sqrt{\frac{b^2}{a}} + \sqrt{2b}\Big)\sqrt{6}\log n$. The case of $r_s \leq 2r_d$ will follow similarly. Now we need the labeling to be successful for n pairs of vertices (so that a spanning tree can be formed). Applying union bound over n distinct pairs guarantees the probability of recovery as $1 - 3/n$. □

Now instead of relying only on the triangle motif, if we consider all different motifs (as defined in Table 2), and then take the aggregate (majority vote) decision, we can improve the above theorem slightly.

Theorem 2. *If* $r_s = \frac{a\log n}{n}$ *and* $r_d = \frac{b\log n}{n}$, *the algorithm considering all three motifs (see Table 2) for a pair of nodes can recover the clusters* V_1, V_2 *correctly with probability* $1 - O(\frac{1}{n})$ *if* $\frac{(3a-2b)(a-2b)}{4a} \geq \sqrt{3}\min\Big\{\sqrt{\frac{3a}{4}} + \sqrt{\frac{b^2}{a}} + \sqrt{2b}, \sqrt{a/2} + \sqrt{\frac{b(a-b)}{a}} + \sqrt{a-b}\Big\}$ *when* $a \geq 2b$, *and* $\frac{(a-b)(a-2b)}{2b} \geq \sqrt{3}\min\Big\{\sqrt{\frac{3a}{4}} + \sqrt{b - \frac{a}{4}} + \sqrt{2a - \frac{a^2}{2b}}, \sqrt{a/2} + \sqrt{\frac{a^2+2b^2-2ab}{2b}}\Big\}$ *when* $a < 2b$.

The proof of this theorem is present in the full version of this paper (Galhotra et al. 2017).

Remark 1. Instead of using Chernoff bound we could have used better concentration inequality (such as Poisson approximation) in the above analysis, to get tighter condition on the constants. We again preferred to keep things simple.

Remark 2 (GBM for $t = 3$ and above)**.** For GBM with $t = 3$, to find the number of common neighbors of two vertices, we need to find out the area of intersection of two spherical caps on the sphere. It is possible to do that. It can be seen that, our algorithm will successfully identify the clusters as long as $r_s, r_d \sim \sqrt{\frac{\log n}{n}}$ again when the constant terms satisfy some conditions. However tight characterization becomes increasingly difficult. For general t, our algorithm should be successful when $r_s, r_d \sim \left(\frac{\log n}{n}\right)^{\frac{1}{t-1}}$, which is also the regime of connectivity threshold.

Remark 3 (More than two clusters)**.** When there are more than two clusters, the same analysis technique is applicable and we can estimate the expected number of common neighbors. This generalization is straightforward but tedious.

Motif counting algorithm for SBM. While our algorithm is near optimal for GBM in the regime of $r_s, r_d \sim \frac{\log n}{n}$, it is far from optimal for the SBM in the same regime of average degree. Indeed, by using simple Chernoff bounds again, we see that the motif counting algorithm is successful for SBM with inter-cluster edge probability q and intra-cluster probability p, when $p, q \sim \sqrt{\frac{\log n}{n}}$. The experimental success of our algorithm in real sparse networks therefore somewhat enforce the fact that GBM is a better model for those network structures than SBM.

5 Experimental Results

In addition to validation experiments in Section 2.1 and 2.2, we also conducted an in-depth experimentation of our proposed model and techniques over a set of synthetic and real world networks. Additionally, we compared the efficacy and efficiency of our motif-counting algorithm with the popular spectral clustering algorithm using normalized cuts[1] and the correlation clustering algorithm (Bansal, Blum, and Chawla 2004).

Real Datasets. We use three real datasets described below.

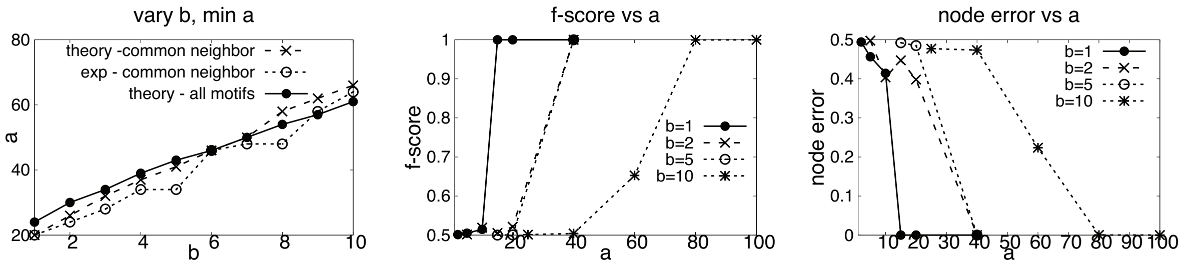
- **Political Blogs.** (Adamic and Glance 2005) It contains a list of political blogs from 2004 US Election classified as liberal or conservative, and links between the blogs. The clusters are of roughly the same size with a total of 1200 nodes and 20K edges.
- **DBLP.** (Yang and Leskovec 2015) The DBLP dataset is a collaboration network where the ground truth communities are defined by the research community. The original graph consists of roughly 0.3 million nodes. We process it to extract the top two communities of size $\sim$ 4500 and 7500 respectively. This is given as input to our algorithm.
- **LiveJournal.** (Leskovec, Adamic, and Huberman 2007) The LiveJournal dataset is a free online blogging social network of around 4 million users. Similar to DBLP, we extract the top two clusters of sizes 930 and 1400 which consist of around 11.5K edges.

We have not used the academic collaboration (Section 2.1) dataset here because it is quite sparse and below the connectivity threshold regime of both GBM and SBM.

Synthetic Datasets. We generate synthetic datasets of different sizes according to the GBM with $t = 2, k = 2$ and for a wide spectrum of values of r_s and r_d, specifically we focus on the sparse region where $r_s = \frac{a\log n}{n}$ and $r_d = \frac{b\log n}{n}$ with variable values of a and b.

Experimental Setting. For real networks, it is difficult to calculate an exact threshold as the exact values of r_s and r_d are not known. Hence, we follow a three step approach. Using a somewhat large threshold T_1 we sample a subgraph S such that u, v will be in S if there is an edge between u and v, and

[1] http://scikit-learn.org/stable/modules/clustering.html#spectral-clustering



(a) Triangle motif varying b and minimum value of a that satisfies the accuracy bound.

(b) f-score with varying a, fixing b.

(c) Fraction of nodes misclassified.

Figure 4: Results of the motif-counting algorithm on a synthetic dataset with 5000 nodes.

Dataset	Total no. of nodes	T_1	T_2	T_3	Accuracy		Running Time (sec)	
					Motif-Counting	Spectral clustering	Motif-Counting	Spectral clustering
Political Blogs	1222	20	2	1	**0.788**	0.53	1.62	**0.29**
DBLP	12138	10	1	2	**0.675**	0.63	**3.93**	18.077
LiveJournal	2366	20	1	1	**0.7768**	0.64	**0.49**	1.54

Table 3: Performance on real world networks

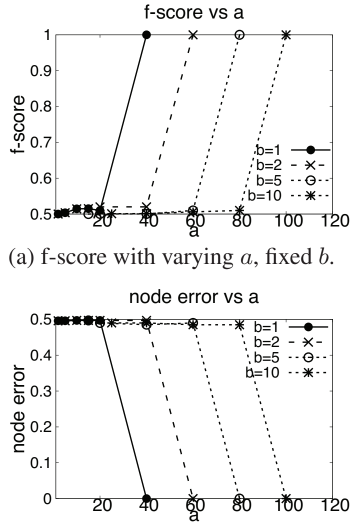
they have at least T_1 common neighbors. We now attempt to recover the subclusters inside this subgraph by following our algorithm with a small threshold T_2. Finally, for nodes that are not part of S, say $x \in V \setminus S$, we select each $u \in S$ that x has an edge with and use a threshold of T_3 to decide if u and x should be in the same cluster. The final decision is made by taking a majority vote. We can employ sophisticated methods over this algorithm to improve the results further, which is beyond the scope of this work.

We use the popular f-score metric which is the harmonic mean of precision (fraction of number of pairs correctly classified to total number of pairs classified into clusters) and recall (fraction of number of pairs correctly classified to the total number of pairs in the same cluster for ground truth), as well as the node error rate for performance evaluation. A node is said to be misclassified if it belongs to a cluster where the majority comes from a different ground truth cluster (breaking ties arbitrarily). Following this, we use the above described metrics to compare the performance of different techniques on various datasets.

Results. We compared our algorithm with the spectral clustering algorithm where we extracted two eigenvectors in order to extract two communities. Table 3 shows that our algorithm gives an accuracy as high as 78%. The spectral clustering performed worse compared to our algorithm for all real world datasets. It obtained the worst accuracy of 53% on political blogs dataset. The correlation clustering algorithm generates various small sized clusters leading to a very low recall, performing much worse than the motif-counting algorithm for the whole spectrum of parameter values.

We can observe in Table 3 that our algorithm is much faster than the spectral clustering algorithm for larger datasets (LiveJournal and DBLP). This confirms that motif-counting algorithm is more scalable than the spectral clustering algorithm. The spectral clustering algorithm also works very well on synthetically generated SBM networks even in the sparse regime (Lei, Rinaldo, and others 2015; Rohe et al. 2011). The superior performance of the simple motif clustering algorithm over the real networks provide a further validation of GBM over SBM. Correlation clustering takes 8-10 times longer as compared to motif-counting algorithm for the various range of its parameters. We also compared our algorithm with the Newman algorithm (Girvan and Newman 2002) that performs really well for the LiveJournal dataset (98% accuracy). But it is extremely slow and performs much worse on other datasets. This is because the LiveJournal dataset has two well defined subsets of vertices with very few intercluster edges. The reason for the worse performance of our algorithm is the sparseness of the graph. If we create a subgraph by removing all nodes of degrees 1 and 2, we get 100% accuracy with our algorithm. Finally, our algorithm is easily parallelizable to achieve better improvements. This clearly establishes the efficiency and effectiveness of motif-counting.

We observe similar gains on synthetic datasets. Figures 4a, 4b and 4c report results on the synthetic datasets with 5000 nodes. Figure 4a plots the minimum gap between a and b that guarantees exact recovery according to Theorem 1 (only triangle motif) and Theorem 2 (all three motifs) vs minimum value of a for varying b for which experimentally (with only triangle motif, and average of three runs) we were able to recover the clusters exactly. Empirically, our results demonstrate the near-optimal performance of motif-counting algorithm, confirming the theoretical bound. We also see a clear threshold behavior on both f-score and node error rate in Figures 4b and 4c. We have also performed spectral clustering on this 5000-node synthetic dataset (Figures 5a and 5b). Compared



(a) f-score with varying a, fixed b.

(b) Fraction of nodes misclassified.

Figure 5: Results of the spectral clustering on a synthetic dataset with 5000 nodes.

to the plots of figures 4b and 4c, they show suboptimal performance, indicating the relative ineffectiveness of spectral clustering in GBM compared to the motif counting algorithm.

References

Abbe, E., and Sandon, C. 2015a. Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery. In *56th Annual Symposium on Foundations of Computer Science (FOCS)*, 670–688. IEEE.

Abbe, E., and Sandon, C. 2015b. Recovering communities in the general stochastic block model without knowing the parameters. In *Advances in Neural Information Processing Systems*, 676–684.

Abbe, E.; Bandeira, A. S.; and Hall, G. 2016. Exact recovery in the stochastic block model. *IEEE Trans. Information Theory* 62(1):471–487.

Adamic, L. A., and Glance, N. 2005. The political blogosphere and the 2004 us election: divided they blog. In *3rd international workshop on Link discovery*, 36–43. ACM.

Avin, C., and Ercal, G. 2007. On the cover time and mixing time of random geometric graphs. *Theoretical Computer Science* 380(1-2):2–22.

Bansal, N.; Blum, A.; and Chawla, S. 2004. Correlation clustering. *Machine Learning* 56(1-3):89–113.

Bettstetter, C. 2002. On the minimum node degree and connectivity of a wireless multihop network. In *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*, 80–91. ACM.

Bollobás, B. 1998. Random graphs. In *Modern Graph Theory*. Springer. 215–252.

Bubeck, S.; Ding, J.; Eldan, R.; and Rácz, M. Z. 2016. Testing for high-dimensional geometry in random graphs. *Random Structures & Algorithms*.

Chin, P.; Rao, A.; and Vu, V. 2015. Stochastic block model and community detection in the sparse graphs: A spectral algorithm with optimal rate of recovery. *arXiv:1501.05021*.

Decelle, A.; Krzakala, F.; Moore, C.; and Zdeborová, L. 2011. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E* 84(6):066106.

Devroye, L.; György, A.; Lugosi, G.; Udina, F.; et al. 2011. High-dimensional random geometric graphs and their clique number. *Electronic Journal of Probability* 16:2481–2508.

Dyer, M. E., and Frieze, A. M. 1989. The solution of some random np-hard problems in polynomial expected time. *Journal of Algorithms* 10(4):451–489.

Galhotra, S.; Mazumdar, A.; Pal, S.; and Saha, B. 2017. The geometric block model. *arXiv preprint* 1709.05510.

Girvan, M., and Newman, M. E. 2002. Community structure in social and biological networks. *Proceedings of the national academy of sciences* 99(12):7821–7826.

Goel, A.; Rai, S.; and Krishnamachari, B. 2005. Monotone properties of random geometric graphs have sharp thresholds. *Annals of Applied Probability* 2535–2552.

Gupta, P., and Kumar, P. R. 1998. Critical power for asymptotic connectivity. In *37th IEEE Conference on Decision and Control*, volume 1, 1106–1110. IEEE.

Haenggi, M.; Andrews, J. G.; Baccelli, F.; Dousse, O.; and Franceschetti, M. 2009. Stochastic geometry and random graphs for the analysis and design of wireless networks. *IEEE Journal on Selected Areas in Communications* 27(7).

Hajek, B. E.; Wu, Y.; and Xu, J. 2015. Computational lower bounds for community detection on random graphs. In *Proceedings of The 28th Conference on Learning Theory, COLT 2015, Paris, France, July 3-6, 2015*, 899–928.

Hajek, B.; Wu, Y.; and Xu, J. 2016. Achieving exact cluster recovery threshold via semidefinite programming. *IEEE Transactions on Information Theory* 62(5):2788–2797.

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

Lei, J.; Rinaldo, A.; et al. 2015. Consistency of spectral clustering in stochastic block models. *The Annals of Statistics* 43(1):215–237.

Leskovec, J.; Adamic, L. A.; and Huberman, B. A. 2007. The dynamics of viral marketing. *ACM Transactions on the Web (TWEB)* 1(1):5.

Leskovec, J.; Lang, K. J.; Dasgupta, A.; and Mahoney, M. W. 2008. Statistical properties of community structure in large social and information networks. In *17th international conference on World Wide Web*, 695–704. ACM.

Mossel, E.; Neeman, J.; and Sly, A. 2015. Consistency thresholds for the planted bisection model. In *47th Annual ACM Symposium on Theory of Computing*, 69–75. ACM.

Penrose, M. D. 1991. On a continuum percolation model. *Advances in applied probability* 23(03):536–556.

Penrose, M. 2003. *Random geometric graphs*. Number 5. Oxford University Press.

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

Yang, J., and Leskovec, J. 2015. Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems* 42(1):181–213.