Community Detection via Discriminant functions for Random Walks in the degree-corrected Stochastic Block Model

Stephen Ragain

December 10, 2017

1 Introduction

Recent work theoretically connected the problem of seed set expansions with personalized page rank by showing that the optimal weights of random walk landing probabilities when constructing a classifier for community detection of nodes in the stochastic block model followed the weights in personalized page rank [3].

One drawback of the stochastic block model is that it can be unrealistic as a model for empirical graphs. The degree-corrected stochastic block model is an extension of the stochastic block model that allows edge probabilities to differ both by which communities the nodes are in as well as by latent "friendliness" parameters that each node has.

The goal of this project was to extend the analysis of [3] to the degree-corrected stochastic block model and determine (i) whether the optimal weights of a classifier using random walk probabilities follow some known node-ranking model (possibly even personalized page rank as with the standard stochastic block model), (ii) a closed form for the weights in terms of the model parameters, and (iii) to explore performance the practicality of such a classifier with experiments.

With this project I was able to extend the analysis of [3] to the degree-corrected stochastic block model, deriving the optimal weights of a classifier in the space of normalized random walk probabilities in terms of the parameters of the model. My central result is a proof akin to the one shown in [3] and highlights that with high probability, random walk probabilities in the degree corrected stochastic block model concentrate around a distribution characterized by a smaller linear system on the blocks with a conditional distribution within blocks proportional to the "friendliness" of each user within that block. This is of independent theoretical interest in relating graph models and random walks (which in turn link to ranking models such as personalized page rank), and in conjunction with some simple parameter estimation was able to produce accurate classification on synthetic data as well as two empirical networks.

2 Related Work

Because this work builds upon a recent result in the literature in style and approach, here I will give a review of the concepts that led to that work as well as why my particular extension to the dcsbm has practical motivation.

The SBM

The *stochastic block model* SBM, is a generative model for random graphs that extends the spirit of Erdos-Renyi $G(n, p)$ graphs to graphs with communities. It presumes the existence of a partition of V into K communities $S_1, \ldots, S_k$ and is parametrized by a $K \times K$ matrix ω giving edge probabilities between clusters. Edge existence is independent for all pairs of nodes, and for each pair $x \in S_i, y \in S_j$ of nodes, the probability $(i, j) \in E$ is ω_{ij}. There are a plethora of results studying the recoverability of $S_1, \ldots, S_k$ given a graph drawn from the SBM under different limits on the number of nodes relative to the values in ω. The SBM is the canonical probabilistic model for community structure in graphs.

Seed set expansion for the SBM via Personalized Page Rank

The *seed set expansion* problem is akin to community detection from a specific node $x-$ supposing there is a community structure underlying a graph, which nodes are x's community.

Both personalized page rank and the heat kernel, two quantities that provide some measure of how connected two nodes x and y are, can both be written as linear combinations of $k-$step random walk probabilities between x and y. A natural question arising from considering these popular ranking models as linear combination of random walk landing probabilities is to wonder which weights are optimal for seed set expansion under a particular random graph model. This is an inversion of sorts on the typical perspective of community detection- given that we wish to use a linear combination of the k-step random walk probabilities from a node x to detect the nodes y in the same community as x, we can compute the optimal weights of those probabilities given a random graph model. This approach was considered by [3], who show that personalized page rank is optimal for the stochastic block model (SBM).

More formally, consider a 2-block SBM, fix some number of steps K, and consider every node y in the graph as a point in $[0,1]^K$ where the k-th r_k^y coordinate of y's vector is it's k step landing probability from some fixed node x. Consider the centroids of the two point clouds of the underlying blocks in this space. *Geometric discriminant functions* $f(r) = w^T r$ place some weights w on features r. If we choose $w = (a - b)$ where a is the centroid of the community containing the seed node and b the centroid of the other community, then $f(r^y) = (a - b)^T y$ gives a ranking of the nodes in terms of their relevance to x, and is normal to the maximal separating hyperplane between the two clouds of nodes in this space. In this sense, this weight vector $w^* = a - b$ gives the optimal weights for classifying the nodes as in the same or difference community as x in the space of the landing probabilities.

Personalized page rank can be written as another discriminant function where the k-th entry is α^k for some $\alpha \in (-1, 1)$ (to interpret these strictly as random walk with α as teleportation probability, simply restrict to $\alpha \geq 0$) and the heat kernel for parameter $t > 0$ has $w_k = \frac{e^{-t}t^k}{k!}$. We refer to these weight vectors as $w_{PPR}(\alpha)$ and $w_{HK}(t)$ respectively. The central result of the paper is that $w_{PPR}(\alpha)$ approaches $b - a$ in the limit of nodes when α is the ratio of the difference and the sum of the edge probabilities from x to its own and the other class. In other words, if p_{in} is the edge probability for pairs of nodes within x's class and p_{out} is the edge probability from x to the other class, for n nodes

$$\lim_{n\to\infty} w_{PPR}\left(\frac{p_{in} - p_{out}}{p_{in} + p_{out}}\right) = b - a.$$

This result extends to more than two communities under mild conditions and for larger classes of classifiers (allowing for correction of correlated random walk probabilities), but the central area for practical improvement of this result, in my mind, is to see it extended to graph models that more often reflect empirical networks.

The Degree-Corrected Stochastic Block Model

When considering networks arising from actual human interaction, the SBM leaves something to be desired. A canonical counterexample to the usefulness of the SBM for community detection is the Karate club graph, where the SBM suggests communities seemingly unrelated to a empirical split of the club and which is intuitively displeasing. A central intuitive issue with the SBM is that under the model the distribution of edges attached to any node $x \in S_i$ is identical– it does not allow for differences between nodes in clusters.

Karrer and Newman set out to extend the SBM so that it more believably reflects empirical graphs, without moving into models such as ERGMs which present tractibility issues in community detection [**?**]. The central problem they correct is that the SBM will sometimes place higher degree nodes in a community opposite the lower degree nodes regardless of any seeming topological structure in the network. The intuitive reason is that by setting diagonal elements of ω near 0 this allows it to use the rest of ω to simply fit to the edge counts of the high-degree and low-degree groups.

The solution proposed combines the Chung-Lu graph model with the SBM, giving every node x a parameter θ_x that controls the degree of x. For $x \in V_I$ and $y \in V_J$, the degree corrected stochastic Block model has

$$P((x, y) \in E | x \in S_i, y \in S_j) = \theta_x \theta_y \omega_{IJ}$$

and all edge's existences are independent. Note that some formulations allow multiedges and have A_{xy} Poisson but with the same expectation. The results here focus on the single edge case but should have no problem extending to that case, which is very similar in sparse regimes.

Because the degree corrected SBM is widespread in empirical applications and has many probabilistc parallels with the SBM (e.g. independent edges), I was able to sucessfully adjust the proof techniques in [3] to the dcsbm.

3 Preliminaries and Notation

We use the notation $G = (V, E)$ to denote a simple, directed graph. A is its adjacency matrix, where A_{xy} is 1 if $(x, y) \in E$. and 0 otherwise. We denote the partition of V into C communities $V_1, \dots, V_C$ with the map $c : V \to \{1, \dots, C\}$ where $c(x)$ is the community node x belongs to.

SBM and DCSBM

The $C \times C$ matrix ω gives the stochastic block model (SBM) edge probabilities. In the traditional SBM, $\{A_{xy}\}$ are independent a bernoulli random variables with probabilities $p_{xy} = \omega_{c(x),c(y)}$ of being 1. We assume that ω is symmetric and for all $I \in \{1, \dots, C\}$ that $\omega_{I,J} < \omega_{I,I}$, so that the probability of an edge between two members of community I is always higher than the probability of an edge between a member of I and another community.

The degree-corrected stochastic block model (DCSBM) also introduces a parameter vector θ indexed by V. For each pair $x, y \in V$ the probability p_{xy} that $(x, y) \in E$ is

$$p_{xy} = \theta_x \theta_y \omega_{c(x),c(y)}.$$

As with the SBM, in the DCSBM, all A_{xy} are inddpendent and bernoulli.

Note that we consider only simple graphs here, a departure from the original work developing the degree-corrected stochastic block model that is for all practical purposes unimportant, as that work focused on Poisson edge counts that were sparse enough to concentrate tightly around our Bernoulli framework.

Linear Discriminant functions

Given an embedding of the nodes of a graph into $\mathbb{R}^K$, consider the centroids a and b of the community of the seed nodes and its complement. A geometric discriminant function $f : V \to \mathbb{R}$ includes a weight vector $w = (b - a)$. If r^v is the embedding of v into $\mathbb{R}^k$, the discriminant function is

$$f(v) = (b - a)^T r^v.$$

The goal here is that the scores of the community containing the seed node and the scores of the rest of the nodes are separated by this function. The body of the paper shows essentially that for SBM, the k-th entry $\hat{w}$ converges to the personalized page rank weights on k-step random walks where the parameter α is a function of the SBM parameters ω.

Fisherian Discriminant Function

Fisherian Discriminant functions work in the same space of features as linear discriminants, but apply a change of variables to the features to maximize class separation. The end result of some linear algebra applied to an optimization formulation is that optimal classification in terms of separation in the new feature space occurs when adjusting the features by the inverse of their covariance Σ:

$$f(v) = (b - a)^T \Sigma^{-1} r^v.$$

Given that we are using random walk probabilities which converge to a stationary distribution for strongly connected graphs (which we are exclusively working with for our problem to be well-defined), this Σ has an interesting interpretation- Σ_{ij} is the correlation between i and j-step landing probabilities from the seed node. In the original space, this gives us a quadratic classifier. Additional illustration of the application and estimation of Σ are discussed later in the report.

4 Theoretical Results

Here we show some analogues of the results from [**?**] that we have derived for the DCSBM problem. The general approach of that paper is do show that for a large enough number of nodes in each community, the number of edges between each pair of communities is concentrated around expectation, and as a result, the number of paths of each length $k = 1, \dots, K$ is concentrated around the solution of a diagonalizable system which has a convenient form.

Although these results do not translate directly to the degree corrected SBM, a similar argument shows that a normalized random walk probability, adjusted by dividing by the degree parameter θ_x of each node x, are tightly concentrated for all nodes in a class around a solution of a linear system. Therefore rather than counting paths directly, we instead focus on a θ correction of these path counts, ultimately leading us to a different feature space for classification.

The first step is to prove the following lemma. A second (simpler) lemma that derives the weights from the linear system found in the first lemma should follow.

Theorem 1 (DCSBM Random Walk Concentration) *Let G_n be an n-node graph with C communities and a distribution $\{\pi_I\}_{I=1}^C$ on the communities with $\pi_I \cdot n$ nodes in community I, and parameters θ and ω giving node-wise and community-pairwise edge propensities respectively.*

For each node x let $c(x)$ be the community of x and let p_x^k be the random walk probability of arriving in node x from some source node s distributed uniformly in community C_0. Then for any $\epsilon, \delta > 0$ there exists some n large enough so that with probability at least $1 - \delta$,

$$\frac{p_x^k}{\theta_x} \approx \frac{\theta_{c(x)} q_{c(x)}^k}{\sum_J q_J^k}$$

in the sense that p_x^k/θ_x is within a $(1 \pm \epsilon)$ multiplicative factor of the above term, where $\{q_I^k\}$ is the deterministic solution to the linear system

$$q_I^k = \sum_{J=1}^{C} D_{JI} q_J^{k-1}$$

where $D_{JI} = \pi_J \theta_J \omega_{JI}$ for $\theta_J = \sum_{y \in J} \theta_y$, and $q_I^0 = 1(I = C_0)$ is an indicator function for the starting community C_0.

Note that all terms in the expression $\frac{q_{c(x)}^k}{\pi_{c(x)} N \sum_J q_J^k}$ are deterministic and depend only on $c(x)$, not x itself. So we have with high probability that the random walk probabilities normalized by θ concentrate for each community around a linear system that depends only on π,$\{\theta_I\}_{I=1}^C$, and ω.

The proof of this theorem breaks down into two lemmas, the first showing degree concentration between each node and community with high probability, and the second showing that such concentration translates into random walk concentration around the given linear system. The central idea behind the difference is that the distribution of landing in a node in k steps under the SBM is entirely function of its class, so the conditional distribution of the node given the class is uniform. For the dcsbm, however, the conditional distribution is proportional to θ on that class. It follows that for classification purposes the random walk probability normalized by θ give the relevant information for community detection.

4.1 Edge concentration across communities for DCSBM

Lemma 1 (DCSBM edge concentration) *For any $\epsilon, \delta > 0$, there exists a large enough n so that for every node x and community J, the number of edges between x and nodes in J is within a $(1 \pm \epsilon)$ factor of its expectation with probability at least $1 - \delta$.*

Proof: Recalling that each A_{xy} is an independent bernoulli, we consider the following random variables for each $x \in V$ and $J \in \{1, \ldots, C\}$:

$$d_{xJ} = \sum_{y \in V_J} \frac{\theta_y}{\theta_J} A_{yx}$$

where $\theta_I = \sum_x \theta_x$ for any community I. We have that

$$\begin{aligned}\mathbb{E}[d_{xJ}] &= \sum_{y \in V_j} \frac{\theta_y}{\theta_C} E[A_{xy}] \\ &= \omega_{J,c(x)} \theta_x \theta_J^{-1} \sum_{y \in V_j} \theta_y^2\end{aligned}$$

Letting $\mu_{xJ} = \mathbb{E}[d_{xJ}]$, we have by the independence of A_{xy} the following Chernoff Bound for any $\epsilon \in (0, 1]$

$$Pr[d_{xJ} < (1-\epsilon)\mu_{IJ}] < \left(\frac{e^{-\epsilon}}{(1-\epsilon)^{(1-\epsilon)}}\right)^{\mu_{xJ}}$$

and a similar bound holds for $Pr[d_{xJ} > (1-\epsilon)\mu_{xJ}]$. These both admit the weaker bound $\exp(-\mu_{xJ}\epsilon^2/2)$ for the probability of falling in the tail, so a union bound over the $2Cn$ events that all d_{xJ} lie within a multiplicative factor of $1 \pm \epsilon$ of their expectation is

$$Pr[\exists I, J s.t. |d_{IJ}/\mu_{IJ}| > (1-\epsilon)] < 2 \sum_{I \neq J} e^{-\mu_{IJ}\epsilon^2/2} \leq 2Cne^{-\tilde{\mu}\epsilon^2/2}$$

where $\tilde{\mu}$ is the minimum over x, J of $\theta_x \omega_{c(x)J} \theta_J$. It follows that for n large enough so that $\delta < 2Cne^{-\tilde{\mu}\epsilon^2/2}$ so long as $\tilde{\mu}$ doesn't shrink too quickly as n grows.

Noting that $\sum_{y \in V_J} \theta_y^2 \le (\sum_{y \in V_j} \theta_y)^2 = \theta_J^2$, we have that $\tilde{\mu}$ is bounded below by $\min \omega_{J,I} \theta_I \min_{x \in I} \theta_x \ge \min \omega_{JI} n \pi_I \min_x \theta_x$, so More precisely the bound holds when $e^{-\tilde{\mu}} \in o(n)$ or equivalently, $\min_x \theta_x \in \Omega(\log(n))$. This means that the concentration holds so long as the graph doesn't become exponentially relatively sparser as n grows, meaning that we have not buried unreasonable assumptions about density into this condition.

Normalized path counting

Lemma 2 *When all d_{xJ} are within $(1 \pm \epsilon)$ of their expectation, the number of paths P_x^k of length k to node x from the source node is within $(1 \pm \epsilon)^k$ of*

$$\frac{\theta_x}{\theta_{c(x)}} q_{c(x)}^k$$

where for all I, k, $q_I^k = \sum_J q_J^k D_{JI}$ for $D_{IJ} = \theta_I \omega_{IJ} \sum_{y \in J} \theta_y^2$. and $q_I^0 = 1(I = C_0)$ where C_0 is the community from which the source node is selected uniform at random.

Proof:

We proceed by induction on k and show the lower bound first. Assume the concentration of the d_{xJ} hold. The base case $k = 0$ is trivially satisfied by the condition $q_I^0 = 1(I = C_0)$. We assume the inductive hypothesis for natural numbers up to k and consider the random variables P_x^k to be the number of length k paths from the source node to x.

$$P_x^{k+1} = \sum_J \sum_{y \in J} P_y^k A_{yx}$$

using the inductive hypothesis we have

$$\begin{aligned}
\sum_J \sum_{y \in J} P_y^k A_{yx} &\ge (1-\alpha)^k \sum_J \sum_{y \in J} \frac{\theta_y}{\theta_J} q_J^k A_{yx} \\
&= (1-\alpha)^k \sum_J q_J^k \sum_{y \in J} \frac{\theta_y}{\theta_J} A_{yx} \\
&\ge (1-\alpha)^k \sum_J q_J^k (1-\alpha) d_{xJ} \\
&= (1-\alpha)^{k+1} \frac{\theta_x}{\theta_I} \sum_J q_J^k d_{JI} \\
&= (1-\alpha)^{k+1} \frac{\theta_x}{\theta_I} q_I^{k+1}
\end{aligned}$$

where we have used the recursive definition of q_I^{k+1} and that for any $x \in I$, $d_{xJ} = \frac{\theta_x}{\theta_I} d_{JI}$. The other direction follows similarly. It follows immeadiately that

Proof of theorem 1: Given ϵ and δ, choose γ such that $\frac{1+\epsilon}{1-\epsilon} > 1 + \gamma$ and $\frac{1-\epsilon}{1+\epsilon} < 1 - \gamma$. Then by Lemma 1 take n large enough so that every d_{xJ} is within $(1 \pm \gamma)^{1/K}$ where of its expectation with probability $1 - \delta$. It follows from lemma 2 that for all x each P_x^k is thus within $(1 \pm \gamma)$ of $\frac{\theta_x}{\theta_{c(x)}} q_{c(x)}^k$ where $q_{c(x)}^k$ is given by the linear system in the theorem statement. To get the p_x^k from P_x^k, we simply divide by $\sum_{x \in V} P_x^k$ with is within a $(1 \pm \gamma)$ factor of $\sum_J \sum_x \theta_x / \theta_J q_J^k = \sum_J q_J^k$. Given that a $1 + \gamma$ distortion in the numerator and a $1 - \gamma$ distortion in the denominator and vice versa both give a distortion of at most $(1 \pm \epsilon)$ in the fraction, the proof is complete.

5 DCSBM-Rank

Here I give the algorithm for detecting the community containing some specified source node s in a directed graph.

1. Estimate $\hat{\theta}$ and $\hat{\alpha}$.

2. For $k = 1, \ldots, K$, where K is given when path counts become numerically cumbersome, compute the landing probability p_x^k of each node x from s in k steps.

3. For each node x compute $r_x = \left(\frac{p_x^1}{\hat{\theta}_x}, \frac{p_x^2}{\hat{\theta}_x}, \dots, \frac{p_x^K}{\hat{\theta}_x}\right)$

4. Estimate $\hat{\Sigma}$ from $\{\{p_x^k\}_x\}_{k=1}^K$.

5. For $\alpha_K = (\hat{\alpha}, \hat{\alpha}^2, \dots, \hat{\alpha}^K)$ output $f = \alpha_K^T \hat{\Sigma}^{-1} r$, with $f(x)$ being the score assigned to node x for lying in the same community of s.

Procedures for parameter estimation are discussed in the next section, but are not tailored to the efficacy of this method.

Parameter Estimates

Because the algorithm divides by each random walk probability by θ_i and uses weights α which depend on the block edge probabilities ω, application of this model to empirical graphs requires estimation of θ and ω.

Note that computation time and accuracy both grow with K, but for K too large estimates become numerically unstable, and the statistical value of landing probabilities in separating classes quickly grows dubious in cases of model-misspecification. In this work I have simply used $K = 4$, though [3] motivates the tracking of the condition number of the matrix of path counts.

In general MLE for the dcsbm is NP-hard, as it subsumes MLE for for the stochastic block model, which can be reduced to graph bisection [?]. While there are estimators for the parameters which work well under assumptions involving e.g. separation between in-class and out-class edge probabilities (keeping the graph far from the notorious resolution limit) such heuristics for estimation can have poor performance outside the model class, which likely includes any empirical network. Furthermore, these estimates can be computationally expensive and noisy, and incremental improvements are unlikely to improve predictive performance when the process generating empirical graphs fell outside the model class anyway. Thus I employ simply moment-based estimation of θ, which amounts quite literally to

$$\hat{\theta}_x = deg(x),$$

as if we fix $||\theta_J||$ for each J, often done so that the MLE is well defined, we have $\mathbb{E}[deg(x)] = \theta_x \sum_J \theta_J \omega_{c(x)J}$, which is a constant (up to the constraint) in θ_x.

To estimate α one can use algorithms such as in [1]. Developing estimators for α is the clear next step in continuous of this work.

Estimation of Σ can be done in a variety of black-box ways, most simply with the sample covariance matrix. In my experiments and visualizations I employed Ledoit-Wolf shrinkage to my covariance matrices. Ledoit-Wolf shrinkage is generally a good idea and I found gave it sharper visuals and significantly improved my classification success on empirical data. Furrther details on Ledoit-Wolf shrinkage can be found in [?].

6 Results on Data

Here we visualize the effect of normalization on the features on synthetic data both in expectation as well as in sampled data. We then explore the classification on an empirical graph arising from webpages for Stanford and Berkeley.

6.1 Visualizations with Synthetic Data

Here I have plotted functions of the random walk probabilities for one through three steps on synthetic graphs drawn from the dcsbm model. The graphs have three communities of 100 nodes each. I consider for two types of graph the random walk probabilities in expectation, the random walk probabilities in expectation dividing by θ_x for the nodes, and the random walk probabilities of a random sample both unprocessed and with the correction by θ and change of basis $\hat{\Sigma}^{-1}$. Note that the visualization in expectation is more illustrative because of the granularity of a one step landing probability, even when scaling by θ_x. My choice of scaling by θ_x is somewhat motivated by my machinations with the theoretical aspect of the project, I may suggest these normalized random walk probabilties as superior features for classification in my final version.

The first graph is modeled after a sort of cycle of the 5 communities, I have included ω for reference, and the θ within each community matches the pattern of any row of ω. The second was obtained by adding a small amount of uniform random noise to a diagonally dominant ω, and by choosing θ to be .75 plus a small amount of uniform at random noise.

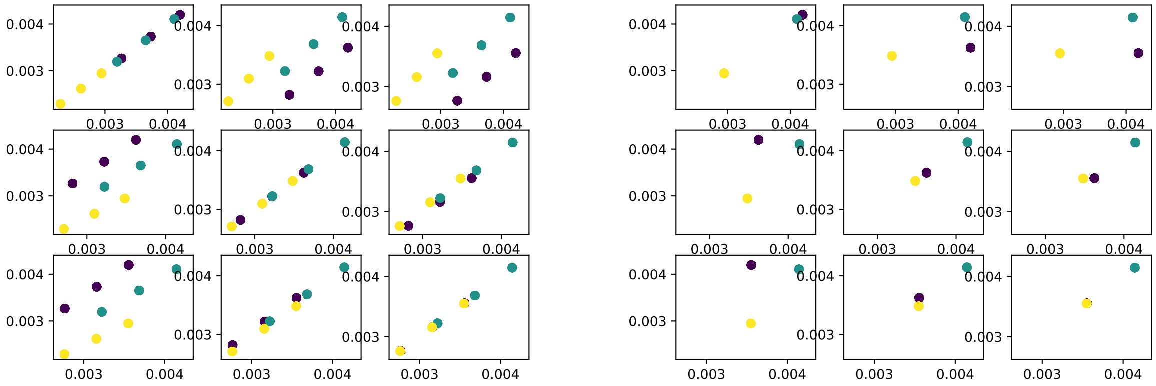

Figure 1: Plots of random walk probabilities in expectation (left) and in expectation when correcting by θ (right) for two synthetic dcsbm graphs. The i, j-th plot from the top left contains the scatterplot of i and j-step random walk probabilities, showing the gaps between community centroids along that plane. Dots represent nodes and are colored by community.

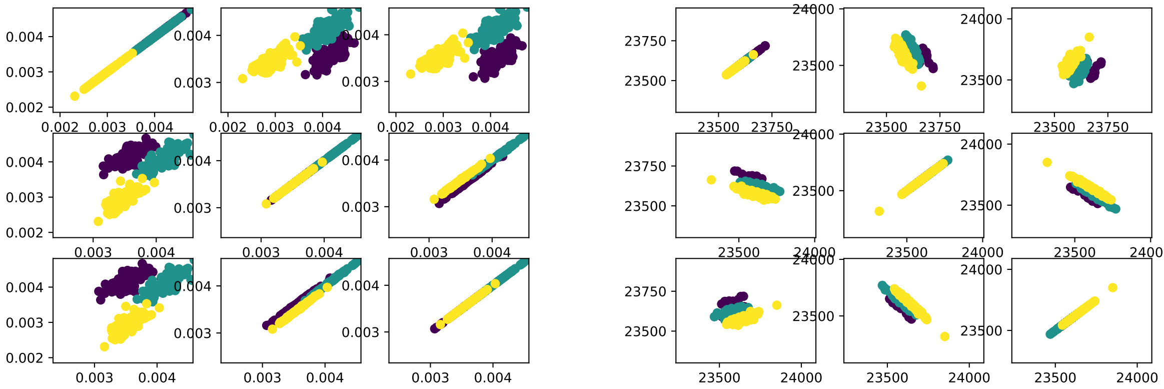

Figure 2: Plots of sampled random walk probabilities and when correcting by θ and $\hat{\Sigma}^{-1}$ (right) for two synthetic dcsbm graphs. The i, j-th plot from the top left contains the scatterplot of i and j-step random walk probabilities, showing the gaps between community centroids along that plane. Dots represent nodes and are colored by community.

We can see that while the random walk probabilities in expectation are reasonably separated point clouds that these contract to single points under the normalization by θ, which gives much cleaner linear separation in this space. We likewise see that while point clouds have nice separation in the sample visually, the degree correction and application of $\hat{\Sigma}^{-1}$ gives sharp linear boundaries between each pair of communities, where we would have had to use non-linear boundaries in the original space.

6.2 Empirical Data Analysis

For my empirical data analysis, I consider the Stanford Berkeley webhosting graph, details of which can be found in [?]. The data consists of webpages with a `stanford.edu` or `berkeley.edu` as nodes and with hyperlinks as directed edges. Then that graph was contracted to its to host domains e.g. `cs.stanford.edu` within each community. The graph has $n = 464$ nodes, 200 corresponding to Berkeley servers and 246 corresponding to Stanford servers.

Evaluation

In order to evaluate my ranking f of nodes, I fixed the community of the 0 indexed node as C_0 and used $s(x) := f(x)/\max_y f(y)$ as a score s for $f(x)$. This means that the lower score a node has, the more likely it is to belong to C_0 under the classification model. For each threshold $t \in [0, 1]$, this allows us to evaluate the classification algorithm $\hat{C}_0 = \{x \in V : s(x) \leq t\}$, $\hat{C}_1 = \hat{C}_0^C$ for precision and recall. These are plotted in figure 3 as a standard Receiver-Operator Curve (ROC). As t moves from 0 to 1, we move along one of the colored curves in that figure, and the point (x, y) tells us that when $\hat{C}_0$ contains proportion of the true C_0, proportion y of the nodes in $\hat{C}_0$ are really from C_0. Thus the area under the ROC curve (AUC) gives us some idea of how our classifier blends high precision and high recall. Barring some difference in utility between false positives and false negatives, it is considered reasonable to judge classifiers monotonically based on their AUC.

We see in the figure that our method improves upon SBM classifier developed in [3]. Here we have used $K = 4$ and $\alpha = .85$ heuristically because of time constraints, but the gap in AUC between dcsbm and sbm was consistently around .075, and widened whenever AUC rose from increasing K or twiddling α.

In particular the performance of FDA under the DCSBM, which has an AUC of .85, is encouraging because it is agnostic to community size and number of blocks and because the parameters are simple estimators or heuristics that have not been optimized.

We find furthermore that the use of FDA rather than LDA, which allows for quadratic boundary shapes in the feature space, improves classification markedly for the degree-corrected model while reducing performance for the pure SBM. A simple explanation here is that in this context, SBM is inappropriate to the extent that the more nuanced classifier is simply overfitting to the model misspecification, while the more reasonable dcsbm is able to take advantage of wider boundary range.

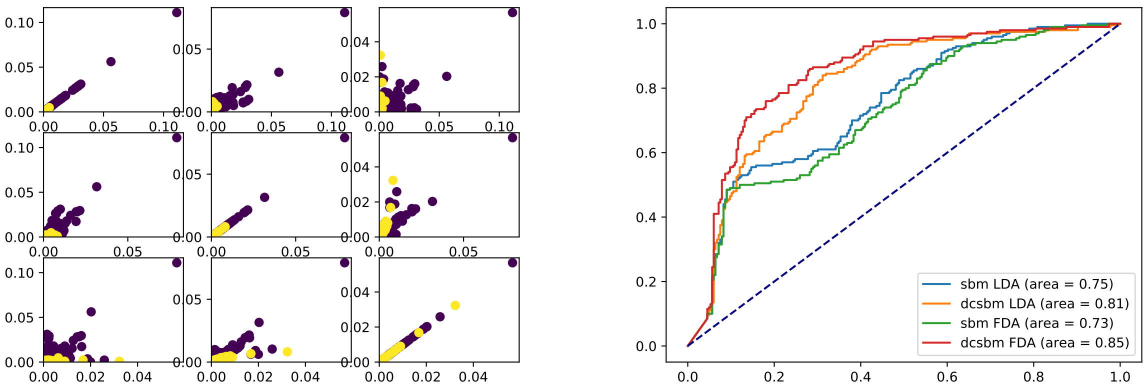

Figure 3: Left: Random walk probabilities in the `Berkstan` graph for 1 through 3 steps. Hosts are are colored by community (Stanford or Berkeley). Right: ROC curve for linear discriminant analysis (LDA) and Fisherian discriminant analysis (FDA) for the the stochastic block model (sbm) and the degree corrected stochastic block model (dcsbm).

7 Conclusion and Continuation

I was able to extend the proofs I wanted to to the model class I wanted to, found a pleasing intuition explaining those proofs, and get some positive results classifying an empirical network with no knob twiddling using a rather agnostic algorithm. Obvious areas for improvement are development of better estimators. I also think that the proofs in this work, which followed a template by the inspiring work, can be both streamlined in terms of elegance as well as generalized to a large class of models centered around edge independence. I will continue working on these problems and hope to have a publishable result in the near future.

Work Breakdown

All work was my own (Stephen's). Thanks for reading!

References

[1] Elizabeth S Allman, Catherine Matias, and John A Rhodes. Parameter identifiability in a class of random graph mixture models. *Journal of Statistical Planning and Inference*, 141(5):1719–1736, 2011.

[2] Bahman Bahmani, Abdur Chowdhury, and Ashish Goel. Fast incremental and personalized pagerank. *Proceedings of the VLDB Endowment*, 4(3):173–184, 2010.

[3] Isabel M Kloumann, Johan Ugander, and Jon Kleinberg. Block models and personalized pagerank. *Proceedings of the National Academy of Sciences*, page 201611275, 2016.

[4] Arjun S Ramani, Nicole Eikmeier, and David F Gleich. Coin-flipping, ball-dropping, and grass-hopping for generating random graphs from matrices of edge probabilities. *arXiv preprint arXiv:1709.03438*, 2017.