Automatic feature classification for the stochastic block model

Lawrence Tray I. Kontoyiannis

May 11, 2021

Abstract

We produce vast quantities of graphical data every day and yet the techniques used to analyse their structure are still in their infancy. We begin with the Stochastic Block Model (SBM) widely used in academia and derive a theorem to verify structure in a labelled graph through a hypothesis test. We apply this theorem to Facebook egonets and find that gender does indeed influence how friendships form on Facebook. Nevertheless, this binary yes-no result is unsatisfactory.

We instead approach the problem from the opposite direction: given a graph we detect the partitions that are in some sense typical and then develop a classifier to map from vertex features to partition. The classifier then implicitly rank orders the input features by importance. Nevertheless, work must still be done to refine this classifier and tie it back to the rigorous hypothesis testing framework already developed.

1 Introduction

There is a wealth of graphical data in the world with more produced every second; social networks, website hyperlinks and academic collaboration are just some examples. Many algorithms are used to analyse this data. Nevertheless, the same principled hypothesis testing framework we have for querying classical data is less well-developed for graphical data. Do my friends vote the same way I do or do researchers collaborate with those of the same gender? We want to answer these questions and not only that, we wish to report our confidence in the answers. To that end we seek to expand the hypothesis testing framework to graphs.

2 The Stochastic Block Model

The most popular graphical model in academia is called the Stochastic Block Model (SBM). We use a definition adapted from Abbe [1].

Definition 2.1 (Canonical SBM) *Let $n \in \mathbb{Z}^+$ be the number of vertices and $k \in \mathbb{Z}^+$ be the number of communities. We define the vector $p = [p_1, p_2 \dots p_k]^T$ to be the prior on the k-communities. Each vertex $v \in \mathcal{V} = \{1, 2 \dots n\}$ has a community label $X_v \in \{1, 2 \dots n\}$. Let W be a symmetric $k \times k$ matrix with entries in $[0, 1]$ called the connectivity matrix. The pair $(X, \mathcal{G}) \sim SBM(n, p, W)$ if X is an $n \times 1$ vector with each component independently drawn from the community prior $X_v \sim p$ and $\mathcal{G}$ is an n-vertex graph where each vertex-pair (i, j) is connected with probability $\mathbb{P}(i \leftrightarrow j) = W_{X_i, X_j}$ independently of other edges. Lastly, we define the community sets as $\Omega_i = \Omega_i(X) \coloneqq \{v \in \mathcal{V} : X_v = i\}$ which contains all vertices belonging to community i.*

3 Verifying Structure

3.1 Theory

Armed with definition 2.1 we tackle the simplest problem in structure verification. Given an undirected graph $\mathcal{G}$ and vertex-labels X such that $(X, \mathcal{G}) \sim \text{SBM}(n, p, W)$, we wish to determine whether two communities a and b connect differently. Put formally, this is a hypothesis test on the parameters of W. There are three parameters we would wish to test: W_{aa}, W_{ab} and W_{bb} (note that for an undirected graph $W = W^T$ necessarily so $W_{ab} = W_{ba}$). To do this we can perform three-pairwise hypothesis tests. Here we test W_α against W_β where α and β are unique index pairs in $\{(a, a), (a, b), (b, b)\}$. Therefore, each hypothesis test can be formulated as below:

$$\begin{aligned} H_0 : &\quad W_\alpha = W_\beta \\ H_1 : &\quad W_\alpha \neq W_\beta \end{aligned} \tag{1}$$

We formulate this as a likelihood ratio test. Letting $\mathcal{L}(D|H)$ denote the likelihood of observing the data $D = (X, \mathcal{G})$ under hypothesis H. Therefore, the test statistic is given by:

$$t_n \coloneqq \log \frac{\mathcal{L}(D|H_1)}{\mathcal{L}(D|H_0)} \tag{2}$$

At this point it helps to introduce some more notation. We define the number of vertices in community i by $n_i \coloneqq |\Omega_i(X)|$. Furthermore, we use $E_{ij} = E_{ij}(X, \mathcal{G})$ to denote the number of realised edges between communities i and j (in generality i may be equal to j) and similarly define $M_{ij} = M_{ij}(X)$ as the maximum number of possible edges between communities i and j. This is computed simply as follows:

$$M_{ij} = M_{ij}(X) = \begin{cases} n_i n_j & \text{for } i \neq j \\ \frac{1}{2} n_i (n_i - 1) & \text{for } i = j \end{cases} \tag{3}$$

With this new notation, the likelihood function can be written explicitly:

$$\begin{aligned} \mathcal{L}(D|H) &= \mathbb{P}(X|p) \cdot \mathbb{P}(\mathcal{G}|W, X) \\ &= \mathbb{P}(X|p) \prod_{i=1}^{k} \prod_{j=i}^{k} \mathbb{P}(E_{ij}|W, X) \\ &= \mathbb{P}(X|p) \prod_{i=1}^{k} \prod_{j=i}^{k} W_{ij}^{E_{ij}} \left(1 - W_{ij}\right)^{(M_{ij} - E_{ij})} \end{aligned} \tag{4}$$

By inspecting equation 4 we see that only terms involving W_α and W_β are going to differ under the two hypotheses H_0 and H_1; the rest of the terms will cancel in our calculation of the test-statistic t_n. Therefore, we can rewrite the likelihood as follows:

$$\mathcal{L}(D|H) \propto f(W_\alpha, E_\alpha, M_\alpha) \cdot f(W_\beta, E_\beta, M_\beta) \tag{5}$$

$$\text{where } f(w, e, m) \coloneqq w^e \cdot (1 - w)^{(m - e)} \tag{6}$$

We note that $f(w, e, m)$ is simply the probability of observing a specific sequence of e successes in m independent Bernoulli trials with parameter w. Its maximiser with respect to the first argument is easily computed through partial differentiation giving:

$$\arg\max_w f(w, e, m) = \hat{w} = e/m \tag{7}$$

Furthermore, we spot the following property $f(w, e_1, m_1) \cdot f(w, e_2, m_2) = f(w, e_1 + e_2, m_1 + m_2)$ or in other words, the function f is linear in its second and third arguments given the same first argument. As such, we can manipulate equation 2 greatly to give:

$$\begin{aligned} t_n &= \log \frac{\max_{W_\alpha \neq W_\beta}(f(W_\alpha, E_\alpha, M_\alpha) \cdot f(W_\beta, E_\beta, M_\beta))}{\max_{W_\alpha = W_\beta}(f(W_\alpha, E_\alpha, M_\alpha) \cdot f(W_\beta, E_\beta, M_\beta))} \\ &= \log \frac{\max_p f(p, E_\alpha, M_\alpha) \cdot \max_q f(q, E_\alpha, M_\alpha)}{\max_r f(r, E_\alpha + E_\beta, M_\alpha + M_\beta)} \\ &= \log \frac{f(\hat{p}, E_\alpha, M_\alpha)}{f(\hat{r}, E_\alpha, M_\alpha)} + \log \frac{f(\hat{q}, E_\beta, M_\beta)}{f(\hat{r}, E_\beta, M_\beta)} \end{aligned} \tag{8}$$

Where $\hat{p} \coloneqq E_\alpha/M_\alpha$, $\hat{q} \coloneqq E_\beta/M_\beta$ and $\hat{r} \coloneqq (E_\alpha + E_\beta)/(M_\alpha + M_\beta)$. These are essentially max-likelihood estimates of the parameters of W constrained by the corresponding hypothesis. Note that the max-likelihood estimate of W_α under H_1 is denoted by the symbol $\hat{p}$ and should not be confused with the community prior p though context will make the distinction apparent. We now state lemma 3.1.

Lemma 3.1 (KL divergence) *With f defined as in equation 6, $0 \leq e \leq m$ and $r \in [0, 1]$ it holds that:*

$$\log \frac{f(e/m, e, m)}{f(r, e, m)} = m \cdot \mathcal{D}\left(Bern(e/m) || Bern(r)\right)$$

where $\mathcal{D}(g||h)$ is the Kullback-Leibler divergence between two probability mass functions (pmf's) $g, h : \mathcal{X} \mapsto [0, 1]$. $\mathcal{D}$ is defined in discrete space as $\mathcal{D}(g||h) \coloneqq \sum_{x \in \mathcal{X}} g(x) \log \frac{g(x)}{h(x)}$ and Bern(p) denotes the Bernoulli pmf with parameter p.

Proving lemma 3.1 is simply a case of algebraic manipulation and is omitted here. The lemma allows us to simplify the test-statistic into a form that is easier to compute (equation 9).

$$\begin{aligned} t_n = &M_\alpha \mathcal{D}\left(\text{Bern}(\hat{p}) || \text{Bern}(\hat{r})\right) \\ &+ M_\beta \mathcal{D}\left(\text{Bern}(\hat{q}) || \text{Bern}(\hat{r})\right) \end{aligned} \tag{9}$$

However, we must simplify further using lemma 3.2.

Lemma 3.2 (χ^2 approximation) *For $g(x) \approx h(x)$ then $\mathcal{D}(g||h) \approx \frac{1}{2}\chi^2(g||h)$ where χ^2 is the chi-squared distance between two distributions defined as $\chi^2(g||h) \coloneqq \sum_{x \in \mathcal{X}} \frac{(g(x) - h(x))^2}{h(x)}$*

The proof requires defining $\delta(x) \coloneqq g(x) - h(x)$ and taking a Taylor expansion of $\log(1 + \delta/h)$ - omitted for brevity. The chi-squared distance between two Bernoullis $g = Bern(p)$ and $h = Bern(q)$ has a very simple form:

$$\chi^2(\text{Bern}(p) || \text{Bern}(q)) = \left(\frac{p - q}{\sqrt{q(1 - q)}} \right)^2 \tag{10}$$

We can apply these results to equation 9 to obtain an approximate expression for t_n valid for large n under the null hypothesis H_0. Under these conditions, the estimates $\hat{p}$, $\hat{q}$ and $\hat{r}$ are all close meaning Lemma 3.2 is applicable. This gives:

$$\begin{aligned} t_n &\approx \frac{1}{2\hat{r}(1 - \hat{r})} \left(M_\alpha (\hat{p} - \hat{r})^2 + M_\beta (\hat{q} - \hat{r})^2 \right) \\ &= \frac{1}{2} \left(\frac{\hat{p} - \hat{q}}{\sqrt{\hat{r}(1 - \hat{r})(1/M_\alpha + 1/M_\beta)}} \right)^2 = \frac{1}{2} z_n^2 \end{aligned} \tag{11}$$

Through application of the Central Limit Theorem (CLT) and Slutsky's Theorem it can be shown that as $n \to \infty$ under the null H_0, the term in brackets on the final line is distributed like a standard Gaussian $z_n \sim \mathcal{N}(0, 1)$. The proof is long-winded and therefore omitted from this short report. The result leads to theorem 3.3.

Theorem 3.3 (χ^2 test) *For $(X, \mathcal{G}) \sim SBM(n, p, W)$, given the realised graph and class labels $(X, \mathcal{G})$ we can perform a hypothesis test on entries W_α and W_β of the connectivity matrix W:*

$$\begin{aligned} H_0 &: \quad W_\alpha = W_\beta \\ H_1 &: \quad W_\alpha \neq W_\beta \end{aligned}$$

If the log-likelihood ratio test statistic t_n is computed as in equation 9, then as the number of vertices $n \to \infty, t_n \sim \frac{1}{2}\chi_1^2$ under the null H_0. Therefore, we reject H_0 at the $100(\zeta)\%$ confidence level if and only if $2t_n \geq \psi^{-1}(\zeta)$, where ψ^{-1} is the χ_1^2 inverse cdf satisfying $\mathbb{P}(Y \leq \psi^{-1}(\zeta)) = \zeta$ given $Y \sim \chi_1^2$.

Indeed, a corollary is the hypothesis test possible by comparing z_n to a standard Gaussian.

3.2 Early results

We seek to apply theorem 3.3 to real-world graphical datasets. We start by analysing social network graphs. The Stanford Network Analysis Project (SNAP) [5] offers a wealth of Facebook egonets. An egonet is simply a graph where all vertices (in this case Facebook users) are connected to one central node (the egonode). The data consists of the undirected set of edges $\mathcal{G}$, indicating whether any two vertices are connected (friends on Facebook) and matrix F, containing binary feature flags for each vertex. Example feature flags are given below:

Example anonymised feature flags

```
75 first_name;anonymized feature 75
76 first_name;anonymized feature 76
77 gender;anonymized feature 77
78 gender;anonymized feature 78
79 hometown;id;anonymized feature 79
```

Each feature is given an anonymised signature to avoid disclosing personally identifiable information. If we have a total of f features and n vertices, then the feature matrix F would have shape $n \times f$ where each row is the feature vector for the corresponding vertex. Each feature is encoded with a binary flag such that $F_{ij} \in \{0, 1\}$ indicating feature turned off and on respectively.

We perform a hypothesis test in the manner described by theorem 3.3 to determine whether gender influences how friends connect on Facebook. We choose to analyse SNAP egonet with id 0 (the id of the egonet is the id of the single egonode). The egonet is plotted on figure 1 with nodes coloured by gender. We use the Python package NetworkX [4] for its graph visualisation tools. Indeed, we will discuss layout algorithms later as for now we focus on the hypothesis test.

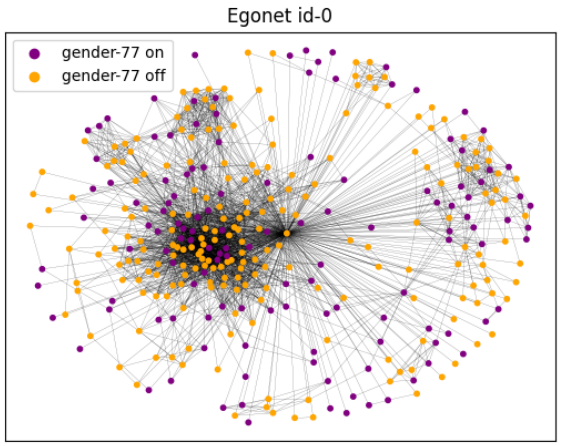


Figure 1: Egonet 0 with nodes coloured by gender

We use an SBM with $k = 2$ communities (1: gender-77 on and 2: gender-77 off) to model the egonet and perform a three-way hypothesis test on the parameters of the connectivity Matrix W.

$$W = \begin{bmatrix} p_1 & q \\ q & p_2 \end{bmatrix} \tag{12}$$

Therefore, p_1 is the probability that two vertices of gender-77 are connected, q is the crossover probability and p_2 is the connection probability within the gender-78 community [1] . The results of the hypothesis tests are given in table 2 alongside p-values (we choose a 95% significance level to reject the null). However, in some cases the test statistics were so extreme that the p-value saturated to 0.

n_1	n_2	$\hat{p_1}$	$\hat{q}$	$\hat{p_2}$
130	218	8.13×10^{-2}	7.66×10^{-2}	10.6×10^{-2}

Table 1: Egonet-0 properties and parameter estimates

Test	H_0	H_1	p-value	H_0 rejected
1	$p_1 = q$	$p_1 \neq q$	0.158	No
2	$p_1 = p_2$	$p_1 \neq p_2$	0.000	Yes
3	$q = p_2$	$q \neq p_2$	0.000	Yes

Table 2: Egonet-0 hypothesis test results

Test 2 rejecting the null gives evidence that gender-78 people (community 2) have on average more same-gender friends than 77-ers have ($\hat{p_2} > \hat{p_1}$ in table 1). The rejection of the null in test 3 gives evidence that gender-78 treat gender-77 people differently to their own gender. Indeed, a gender-78 person is more likely to be friends with a fellow 78-er than with a 77-er ($\hat{p_2} > \hat{q}$).

However, we do not reject the null in test 1 as there is insufficient evidence to claim the reverse (that 77-ers tend to stay away from 78-ers). Nevertheless, these are results on a single egonet so can hardly be generalised to society as a whole.

That said, the exercise has highlighted some interesting points. The hypothesis test framework does not quantify the magnitude of the difference just whether or not a difference exists. Furthermore, The method relies on the analyst to specify the features of interest so results may be misleading if third variables are missing from the analysis.

To address these shortcomings we instead approach the problem from a different angle. For now we have been given a graph with fully labelled vertices and ask if a given label affects the structure. Instead, we could take a graph and partition first (without using feature information) and then ask which features explain the partition. For that we need a way of detecting structure.

4 Detecting Structure

4.1 Weak recovery

Community detection is the problem of recovering the communities from the graph $\mathcal{G}$. There are 4 recovery regimes, as defined by Abbe [1]. We choose to focus on weak recovery as it is the least strict of all. Therefore, there are very few constraints on the SBM parameters and the theory can be applied to a wide range of graphs. A statement of weak recovery is given in definition 4.1 (adapted from [1]).

Definition 4.1 (Weak recovery) *weak recovery is solved for a graph $(X, \mathcal{G}) \sim SBM(n, p, W)$ if there exists $\epsilon > 0$, a certain choice of indices $i, j \in \{1, 2 \dots k\}$ and an algorithm that takes as input $\mathcal{G}$ and outputs a partition of the vertex set $\mathcal{V}$ into two distinct sets S and S^C (detected communities 1 and 2 respectively) such that:*

$$\mathbb{P}\left\{ \frac{|\Omega_i \cap S|}{|\Omega_i|} - \frac{|\Omega_j \cap S|}{|\Omega_j|} \geq \epsilon \right\} = 1 - o(1)$$

Where $o(1)$ denotes a family of functions that tend to 0 as n increases. We recall the definition of $\Omega_i = \Omega_i(X) \coloneqq \{v \in \mathcal{V} : X_v = i\}$. In other words, we require the partition $S \subset \mathcal{V}$ to cut across communities such that the fraction of a particular community in partition S with respect to the whole community is different for a choice of two communities.

One of the most promising weak recovery algorithms is called Acyclic Belief Propagataion (ABP). We implement a linearised version adapted from Abbe and Sandon [2]. This is a message-passing algorithm that solves weak recovery in the sense of definition 4.1. The original paper goes into much more detail than we have space to discuss here. Nevertheless, the premise is that one can infer the community of a particular vertex simply from the labels of all its neighbours. The

[1] Strictly speaking this is the not gender-77 community as there are a handful of vertices that do not disclose their gender to Facebook or do not fall into this binary

algorithm is non-deterministic [2] and requires the graph $\mathcal{G}$ as well as two hyper-parameters r (the maximum cyclic length to correct for) and T (the number of iterations).

We implement linearised ABP in Python and apply it to the Facebook egonets described earlier. We found that the hyperparameter settings $r = 3$ and $T = 5$ yielded good results. The algorithm returns the belief σ_v associated with each vertex v. If this value is positive $\sigma_v > 0$ then we assign $v \in S$; S^C contains all vertices with $\sigma_v \leq 0$. As well as the hard partition of $\mathcal{V}$ into S and S^C, the magnitude of σ_v encodes how strongly we believe that this vertex belongs to its assigned partition. We can rescale the positive and negative components independently to obtain the normalised belief $\tilde{\sigma}_v$ such that $\tilde{\sigma}_v \in [-1, +1]$ and no signs are flipped by the rescaling.

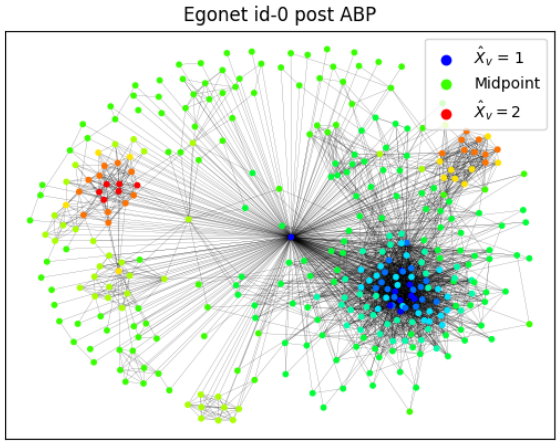


Figure 2: Normalised ABP output $\tilde{\sigma}_v$ colouring vertices, spring layout

We colour each vertex according to this quantity $\tilde{\sigma}_v$ and plot the graph on figure 2. We see intuitively that ABP has worked. The blue cluster in the lower right is clearly part of the same community. Furthermore, many vertices that do not seem to belong to any cluster have uncertain classifications (the colour is close to green). Inspecting figure 1 shows us that the sets S and S^C do separate nicely with high intra-group connectivity and lower inter-group connection density. Indeed, an alternative interpretation of weak recovery is finding the partition that minimises the number of edges between the two sets.

[2] As ABP is non-deterministic, each figure is the result of a different run of ABP. No exact comparison is possible between figures but the results are in some sense typical.

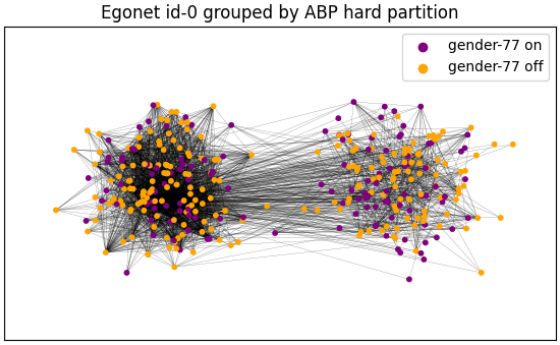


Figure 3: Vertices grouped by ABP partition: set S (left cluster) and set S^C (right cluster)

Within set S we have $\{58, 122\}$ 77-ers and 78-ers respectively and for S^C the breakdown is $\{72, 96\}$. Therefore, the fraction of each gender cohort in set S can be calculated:

$$\begin{aligned} \frac{|\Omega_{77} \cap S|}{|\Omega_{77}|} &= \frac{58}{58+72} = 0.446 \\ \frac{|\Omega_{78} \cap S|}{|\Omega_{78}|} &= \frac{122}{122+96} = 0.560 \end{aligned} \tag{13}$$

Clearly the fractions of each community in the partition S differ substantially so we say that weak recovery is solved in the sense of definition 4.1. The set S contains a higher proportion of the overall 78-population than of the 77-population.

4.2 From detection to inference

We now have an algorithm (ABP) that partitions our graph into two sets that are maximally distinct from a clustering sense. We can now ask the question of which vertex labels most readily explain the separation. Therefore, we can determine which features have the largest impact on the graphical structure; this circumvents the binary resolution problem with the hypothesis testing approach.

This part of the project is still in its early stages so we present preliminary results. We necessarily remove the constraint on the community sets Ω_i to be disjoint as one vertex can have multiple features turned on. We perform a linear regression on the $n \times 1$ normalised belief vector $\tilde{\sigma}$ with the $n \times f$ feature matrix F as the explanatory variable. $F_{ij} = 1$ if vertex i has feature j turned on and 0 if feature j is off. The equation we try to fit is given below:

$$\tilde{\sigma} = a + Fb + \epsilon \tag{14}$$

Linear regression finds the constant bias a and weight vector b that minimises the mean squared prediction error $\epsilon^T \epsilon / n$. The theory for linear regression is well known so omitted here (see [11] for a reference). We restrict F to only include gender and language features and plot the corresponding element of b for each feature in order of decreasing magnitude (figure 4).

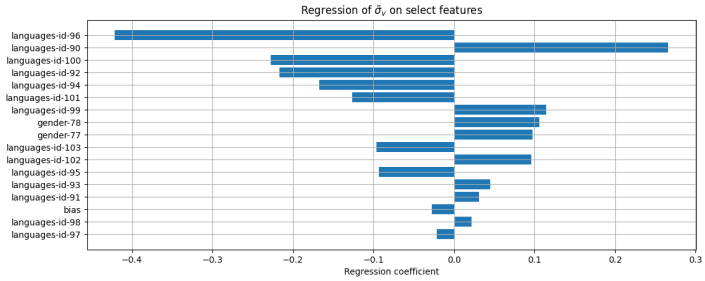


Figure 4: Egonet-0 linear regression of $\tilde{\sigma}_v$ on select features ($R^2 = 0.113$)

The results are rather interesting. Certain languages are a far stronger predictor of community assignment than gender. After all, one would hope that a language barrier poses a more tangible obstacle to friendship than gender. The bias term has rather low magnitude which just implies that the partition S and S^C are of similar size.

This analysis is of course rather crude as we have failed to take into account multi-collinearity or even determined if a linear regression model is appropriate; indeed, an R^2 value of just 0.113 would suggest that it is not. Nevertheless, this serves as the starting point for subsequent more rigorous analysis. Furthermore, we have not yet proven the conjecture that the regression coefficients are valid measures of causal impact on graphical structure.

5 Future Direction

The immediate next steps for the project are to refine the classification approach outlined in section 4.2. Firstly, it should be tied back to the hypothesis testing framework to prove the results are be meaningful. Secondly, the exact form of the classifier can be explored; a linear regression model is a good starting point but we can expand to a simple softmax neural network as that would allow us to generalise to more than just two partitions.

Indeed, ABP is severely limited by the fact that it only outputs 2 sets. Though weak recovery is solved, the partition is often hard to interpret as multiple clusters can get subsumed into one. We instead would prefer to investigate algorithms that can identify arbitrarily many clusters. Force-directed approaches are the standard in industry for their speed of execution. One of the most common is the Fruchterman-Reingold algorithm which treats vertices as repelling objects and edges as restoring forces. It can then perform iterative updates on vertex positions until convergence or termination. This algorithm is what is used by the NetworkX package [4] to compute graph layouts as seen in figures 1 and 2. Song and Bressan [9] have had great success using the Fruchterman-Reingold algorithm followed by k-means clustering for community detection.

Lastly, the analysis must of course be expanded to more datasets to benchmark efficacy against existing methods and ensure the approach is generally applicable. Work has already begun analysing the influence of gender on academic collaboration graphs provided by Aminer [10]. Furthermore, the seminal dataset of political blogs from the 2004 US election [3] which started much of the analysis of the SBM is in the works.

6 Bayesian Exploration

For our inference goals we wish to sample block memberships b according to the posterior $p(b|\mathcal{G})$. Peixoto [7] proposes a Monte-Carlo method for generating these samples. However, he proposes a slight modification of the Stochastic Block Model. The so called microcanonical formulation. In contrast to the canonical version where edge parameters are obeyed only on expectation, the microcanonical model has these constraints obeyed with certainty. The definition follows from [8].

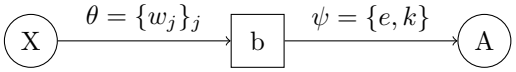
Definition 6.1 (Microcanonical SBM) *Let $N \in \mathbb{Z}^+$ denote the number of vertices in our graph. The block memberships are denoted by a vector b of length N where each entry $b_i \in \{1, 2 \dots B\}$ where B is the number of nonempty blocks. Let e be a $B \times B$ matrix of edge counts between blocks (e_{rs} is number of edges from block r onto block s - or twice that number if $r = s$). We restrict our analysis to undirected graphs so e is symmetric. For a non-degree-corrected stochastic block model (NDC-SBM), we say that the graph A is generated as follows:*

$$A \sim \textit{NDC-SBM}_{MC}(b, e) \tag{15}$$

Where edges are placed at random but respecting the constraints imposed by e and b. The additional parameters N and B are omitted as they are inferred from the shapeofs of b and e. If we interpret A as an adjacency matrix, then this constraint can be written formally as: $e_{rs} = \sum_{i=1}^{N} \sum_{j=1}^{N} A_{ij} \mathbb{1}\{b_i = r\} \mathbb{1}\{b_j = s\}$. However, this formulation does not tolerate high degree variation within blocks as is typical of real-world data. We therefore introduce the degree-corrected SBM (DC-SBM) which has an additional parameter k which is a vector of length N encoding the degree sequence (k_i is the degree of vertex i). Therefore, we write:

$$A \sim \textit{DC-SBM}_{MC}(b, e, k) \tag{16}$$

This imposes the additional constraint that $k_i = \sum_{j=1}^{N} A_{ij}$. In what follows, we will always assume the degree-corrected model unless otherwise specified.



At each iteration of the Markov chain we can use the most recent block assignments to train the classifier, mapping from vertex features to detected membership. The classifier has parameters θ. We wish to apply Bayesian techniques such that we retain a measurement of the uncertainty in the classification. We propose to use Stochastic Gradient Langevin Dynamics (SGLD) as described by Nemeth and Fearnhead [6]. This allows us to draw samples from an arbitrary distribution π on θ. We can write π in the following form.

$$\pi(\theta) \propto \exp\left(-U(\theta)\right) \tag{17}$$

We wish to draw samples from, the posterior.

$$p(\theta|b) = \frac{p(\theta)p(b|\theta)}{p(b)} \tag{18}$$

We can generate these samples and then use them to approximate the true posterior:

$$p(b_v = j|\mathcal{G}) \approx \frac{1}{N}\sum_{n=1}^{N} \mathbb{1}\left\{b_v^{(n)} = j\right\} \tag{19}$$

7 Conclusion

We began by fully fleshing out the theoretical foundation of the Stochastic Block Model and hypothesis tests on connectivity parameters. Nevertheless, this approach is too narrow in scope to be a useful tool on its own. We therefore set about the problem from the opposite direction; rather than verify structure we are already given, we detect structure in the graph without using labels and seek to identify which labels best recreate the detected partition. Early results are promising but this approach needs to be given the same rigorous treatment as the hypothesis testing framework. That will be the emphasis of future work.

References

[1] Emmanuel Abbe. "Community Detection and Stochastic Block Models: Recent Developments". In: *Journal of Machine Learning Research* 18.177 (2018), pp. 1–86. URL: http://jmlr.org/papers/v18/16-480.html.

[2] Emmanuel Abbe and Colin Sandon. "Achieving the KS threshold in the general stochastic block model with linearized acyclic belief propagation". In: *Advances in Neural Information Processing Systems*. Ed. by D. Lee et al. Vol. 29. Curran Associates, Inc., 2016, pp. 1334–1342. URL: https://proceedings.neurips.cc/paper/2016/file/6c29793a140a811d0c45ce03c1c93a28-Paper.pdf.

[3] Lada A. Adamic and Natalie Glance. "The political blogosphere and the 2004 US Election". In: *Proceedings of the WWW-2005 Workshop on the Weblogging Ecosystem*. 2005.

[4] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. "Exploring Network Structure, Dynamics, and Function using NetworkX". In: *Proceedings of the 7th Python in Science Conference*. Ed. by Gaël Varoquaux, Travis Vaught, and Jarrod Millman. Pasadena, CA USA, 2008, pp. 11 –15.

[5] Jure Leskovec and Andrej Krevl. *SNAP Datasets: Stanford Large Network Dataset Collection*. June 2014. URL: http://snap.stanford.edu/data.

[6] Christopher Nemeth and Paul Fearnhead. "Stochastic Gradient Markov Chain Monte Carlo". In: *Journal of the American Statistical Association* 116.533 (2021), pp. 433–450. DOI: 10.1080/01621459.2020.1847120. eprint: https://doi.org/10.1080/01621459.2020.1847120. URL: https://doi.org/10.1080/01621459.2020.1847120.

[7] Tiago P. Peixoto. "Efficient Monte Carlo and greedy heuristic for the inference of stochastic block models". In: *Physical Review E* 89.1 (2014). ISSN: 1550-2376. DOI: 10.1103/physreve.89.012804. URL: http://dx.doi.org/10.1103/PhysRevE.89.012804.

[8] Tiago P. Peixoto. "Nonparametric Bayesian inference of the microcanonical stochastic block model". In: *Physical Review E* 95.1 (2017). ISSN: 2470-0053. DOI: 10.1103/physreve.95.012317. URL: http://dx.doi.org/10.1103/PhysRevE.95.012317.

[9] Yi Song and Stephane Bressan. "Force-directed Layout Community Detection". In: *School of Computing* (2013).

[10] Jie Tang et al. "ArnetMiner: Extraction and Mining of Academic Social Networks". In: *KDD'08*. 2008, pp. 990–998.

[11] Yale. *Linear Regression*. URL: http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm.