A Hierarchical Model for Degree-Heterogenous Random Graphs

Ivan Specht and Eric Shen

Stat 236 Project

Summary. We propose a hierarchical model for modelling random graphs with degree heterogeneity, which we refer to as simply "the model". Our formulation is based on the perspective of describing the distribution of node degrees rather than community detection, and can be viewed as a generalization of the conventional degree-corrected block model (DCBM) without community effects, where the block size is 1. In the model, the signal matrix is defined solely by the *degree effects* for each node, and these degree effects are themselves drawn from some latent distribution which we try to estimate. In this report, we review the literature on common extant models for random graphs with degree heterogeneity; formally introduce our model; and run simulations to illustrate the nature of working with this model in practice.

1 Introduction

In Stat 236, within Chapter 3 of the class content we encountered a variety of random graph models for community detection in class of varying levels of complexity, from simple Erdos-Renyi models to DCBM formulations; explored corresponding methods to estimate parameters for these models (e.g. SCORE (Jin 2015)); and discussed applications of these models on real-world network datasets.

Our interest within this project is primarily in formulating a new way to model random graphs, and exploring some of its properties with respect to the kinds of graphs it generates and the tractability of performing inference with this model. Our motivation comes from the phenomenon of *degree heterogeneity*; that is, the fact that in many real-world networks, the distribution of node degrees follows a power-law distribution, with small proportions of nodes being far more "popular". Within the context of the course, degree heterogeneity was identified as a property of networks

inexpressible by stochastic block matrix (SBM) models of random graphs, motivating the creation of DCBM models. We take a different approach, viewing the power-law distribution of node degrees as purely due to the degree effects of each node (without modelling nodes as belonging to communities), with these degree effects in turn following some latent distribution. In this way, it is this latent distribution that completely parameterizes the edge-forming behavior of the graph.

To begin, we provide some context in the area of random graph models via a summary of selected material covered in the class. While lots of the discussion for SBM and DCBM models was motivated by the problem of community detection, our model notably does *not* factor in communities, so we focus more on the details regarding edge formation rather than reviewing community detection. We next introduce our hierarchical model and examine different methods for performing statistical inference on it. We explore via simulations the kinds of graphs our model generates, and how well the inference techniques we describe do in practice. Finally, we offer some brief remarks on the merits, limits, and usefulness of our proposed model.

2 Background

2.1 SBM and DCBM

We closely follow the notation as given in class. We write a graph as $G = (V, E)$, where V is a set of nodes (vertices) and E is a set of undirected edges between nodes. Let $|V| = n$ and the nodes be $\{1, \ldots, n\}$. The edge (i, j) indicates that nodes i, j are connected. Then the adjacency matrix of G, $\mathbf{Y} \in \{0, 1\}^{n \times n}$, is symmetric, and its entries obey $Y_{ij} = Y_{ji} = 1 \iff (i, j) \in E \ \forall i, j$. We do not consider multiple edges between nodes, directed edges, or self-loops, so $Y_{ii} = 0 \ \forall i$. The degree of node i, d_i, is hence its number of neighbors.

We view $\mathbf{Y}$ as a random matrix, and we consider each of its entries (an edge) as being distributed as a Bernoulli random variable. Some models specify this via a *signal matrix* $\mathbf{\Omega} \in [0, 1]^{n \times n}$, which essentially specifies the probabilities of edge generation: $\Omega_{ij} = \Omega_{ji} = P(Y_{ij} = 1) \ \forall i, j$ and $\Omega_{ii} = 0 \ \forall i$ (some models may parameterize the diagonal of $\mathbf{\Omega}$ differently). Our observed data is the adjacency matrix of the actual graph in $\{0, 1\}^{n \times n}$. (Here, we differentiate the Bernoulli random variable Y_{ij} from observed data y_{ij}, but in practice, since typically we work with one graph and thus one observation, the distinction is unimportant.)

In the SBM model, nodes are partitioned into K communities $\mathcal{C}_1, \ldots, \mathcal{C}_K$, and $P(Y_{ij} = 1) = P_{k\ell}$ if $i \in \mathcal{C}_k, j \in \mathcal{C}_\ell$, effectively giving $\binom{K}{2}$ parameters. It follows that the expected node degree is the same within a community; indeed, for any node $i \in \mathcal{C}_k$, it equals $\sum_{j \neq i} \mathbb{E}[Y_{ij}] = \sum_{j \neq i,\, j \in \mathcal{C}_\ell} P_{k\ell} = |\mathcal{C}_k - 1| P_{kk} + \sum_{\ell \neq k} |\mathcal{C}_\ell| P_{k\ell}$. In practice, this means SBM does not account for the phenomenon of degree heterogeneity. Degree heterogeneity specifically refers to the empirical observation that the distribution of node degrees often follows a power law, i.e. real-world networks are *scale-free*, wherein the fraction of nodes with degree k is distributed proportional to $k^{-\lambda}$ for some constant λ. Degree heterogeneity and scale-free networks are common on real-world network datasets, with one possible explanation for this phenomenon being preferential attachment, wherein newly-added nodes are more likely to form edges to high-degree nodes, spurning the emergence of the power law (Barabási and Albert 1999).

As mentioned, the DCBM model provides a way to add degree heterogeneity to the SBM model. One formulation seen in class is to add individual degree effects for each node to model $\mathbf{\Omega}$. (Karrer and Newman 2011) consider the setting where each node i is given an additional degree effect parameter θ_i with $\Omega_{ij} = \theta_i \theta_j \cdot P_{k\ell}$ if $i \in \mathcal{C}_k, j \in \mathcal{C}_\ell$, and give a formulation for the more general case of undirected multigraphs with self-edges. This introduces an additional vector of parameters $\boldsymbol{\theta}$.

However, the primary goal of DCBM is arguably still for modelling and performing inference on community detection; for instance, the SCORE algorithm provides a way to estimate the community parameters $P_{k\ell}$ without having to estimate the degree effects θ_i, which are considered as "nuisance", by normalizing the eigenvectors of the observed adjacency matrix (Jin 2015). Moreover, it quickly follows that the expected degree of node $i \in \mathcal{C}_k$, d_i, is $\mathbb{E}\left[\sum_{j \neq i} A_{ij}\right] = \theta_i \sum_{j \neq i,\, j \in \mathcal{C}_\ell} \theta_j P_{k\ell}$. This cannot be simplified, for the DCBM model does not make any restrictions on the properties of the θ_i meaning they may be arbitrarily chosen. In contrast, our model is chiefly concerned with describing degree effects rather than communities, for which it makes more sense to focus on more natural forms to describe node degrees.

2.2 Other Random Graph Models

The DCBM above is far from the only extension of SBM motivated by describing degree heterogeneity. (Lee and Wilkinson 2019) reviews different formulations of degree-corrected block models, including for the more general case of multigraphs. (Lu and Szymanski 2019) extend the DCBM for multigraphs,

proposing the *regularized stochastic block model*. This model specifies two constants for each node i, I_i, O_i, instead of one, with the probability of edge formation between nodes $i \in \mathcal{C}_k$ and $j \in \mathcal{C}_\ell$ as $I_i I_j P_{k\ell}$ if $k = \ell$ and $O_i O_j P_{k\ell}$ otherwise. The authors find that this model performs better in finding *assortative* communities, where nodes of similar degree are more likely to be in the same community. However, these models deal with multigraphs and still focus on community detection.

Some models focus on describing mechanisms of graph formation as an alternate characterization of degree heterogeneity and clustering in graphs. The Watts-Strogatz model provides an alternate formulation of clustering in graphs compared to block models, where nodes are first generated in a lattice-like structure and edges are then shuffled ("rewired") with some probability (Watts and Strogatz 1998). However, this model requires as a parameter the mean node degree K, and (Barrat and Weigt 1999) show that the node degree distribution of this model is a delta function centered at K, meaning this model does not satisfyingly address degree heterogeneity. Some hierarchical graph models, aimed at simulating scale-free network graphs, are essentially algorithms specifying recursive methods of graph evolution, wherein nodes from earlier iterations of the graph are designated as "hubs", the graph structure is replicated, and nodes in the replicated graph are connected to hub nodes, e.g. those introduced by (Noh 2003) and (Ravasz and Barabási 2003). These authors show that graphs generated according to these rules follow power laws in the distribution of node degrees.

Other models for describing degree-heterogenous network graphs similarly consider the iterative evolution of a graph over several time steps, but consider the formation of new edges from added nodes as following a probabilistic process rather than an algorithmic one. The *Barabási-Albert model* iteratively grows the network from m_0 initial nodes by, at each step, adding a new node and connecting it to m different sampled existing nodes, where node i is sampled with probability $\frac{d_i}{\sum_j d_j}$ and j indexes all existing nodes; thus, higher-degree nodes are more likely to be connected, which simulates preferential attachment (Albert and Barabási 2002).

Fitness models additionally prescribe an explicit fitness parameter drawn from some distribution to each node, so the probability of a new node being attached to an existing node is a function f of their degrees and/or fitnesses. Fitnesses for nodes are fixed and higher fitnesses mean a higher likelihood of edge formation, meaning that node fitness is not unlike the degree effect in DCBM. The *Bianconi-Barabási* model stands as a natural extension to the Barabási-Albert model, where fitnesses η_i are drawn i.i.d. from a distribution $\rho(\eta)$. The graph grows similarly to the Barabási-Albert

model, except when connecting a new node, node i is sampled with probability $\frac{\eta_i d_i}{\sum_j \eta_j d_j}$, so the Barabási-Albert model can be seen as a degenerate case when all η_i are equal (Barabási 2000). (Servedio, Caldarelli, and Buttà 2004) show general technical conditions under which the distribution of fitnesses and properties of f lead to the formation of scale-free networks.

The Barabási-Albert model, along with fitness model formulations, do not however lend themselves easily to performing inference. While their manner of node and edge generation is stochastic, the use of these models appears tailored to creating, simulating, and studying degree-heterogenous network graphs, rather than trying to fit real-world data. To our knowledge, there is no literature relating to how, given data for a real-world graph, estimates for parameters for the aforementioned models in this section can be found, if we try to approximate the graph with such models. Given this review, the goal at hand is thus to explore a model expressing degree heterogeneity upon which inference can be performed.

3 Proposed Hierarchical Model and Inference

3.1 Model Description

Our aim is to formulate a model that can express degree heterogeneity and does so in a conceptually-simple way that can be fit to datasets through statistical inference. We formally present our model before evaluating its actual usefulness when it comes to inference and the kinds of graphs it generates.

As motivation, consider a setting of individuals absent communities. Some individuals may be more "popular" than others, which we parameterize with degree effects $\theta_i \in [0, 1]$. Then edge formation is determined purely by degree effects: the edge (i, j) where $i \neq j$ is generated with probability $\theta_i \theta_j$ (there are no self-edges). That is, $Y_{ij} \sim \text{Bern}(\theta_i \theta_j)$ and the Y_{ij} are independent.

At this point, we remark that the model is a special case of the DCBM with trivial community size 1. Indeed, with community size 1 the DCBM gives $P(Y_{ij} = 1) = \theta_i \theta_j \cdot P_{ij}$, considering i, j are elements of their respective singleton communities $\mathcal{C}_i, \mathcal{C}_j$. Then for all i, j, reparameterizing $\theta_{\min(i,j)} \leftarrow P_{ij} \theta_{\min(i,j)}$ "absorbs" the community effect parameters P_{ij} into each of the degree effects (here, arbitrarily choosing that it gets absorbed into the effect of the lesser-labeled node, guaranteeing that the value of each P_{ij} is factored into the same θ_i). So without further restrictions on the θ_i, we just get a degenerate case.

To extend our model to a novel and interesting setting, we think it natural to consider the degree effects as themselves drawn from a latent distribution (akin to the initial state of the world). To guarantee that the edge generation probabilities $\theta_i\theta_j$ lie in $[0,1]$, we assume the θ_i are drawn i.i.d. from some probability distribution over $[0,1]$, $\rho \in \Delta_{[0,1]}$. This also lets our model be parameterized in an especially lightweight manner, only requiring the parameters for ρ (in addition to the number of nodes). In this report, we will only consider the Beta distribution, as it is well-known.

Overall, letting $\boldsymbol{\theta}$ indicate the vector of degree effects $(\theta_1, \ldots, \theta_n)$, the model is:

$$\begin{aligned}
\theta_i &\sim \text{Beta}(\alpha, \beta), \quad \text{i.i.d.;} \\
Y_{ij}|\boldsymbol{\theta} &\sim \text{Bern}(\theta_i\theta_j), \quad \text{conditionally i.i.d. given } \boldsymbol{\theta}; \\
\mathbf{Y} &= (Y_{ij}) \quad \text{is the adjacency matrix of graph } G; \\
y_{ij} &\sim Y_{ij} \quad \text{are the observed data samples.}
\end{aligned}$$

Observe that this is a two-stage hierarchical model, and that we only need two parameters (in addition to the number of nodes) to specify our model. We can also readily form the inference problem of, given a generated graph (or graph dataset to model), figuring out the underlying parameters of the Beta (α and β).

3.2 Degree Heterogeneity

Since our aim is for the model to capture degree heterogeneity, we examine how this formulation performs therein. Since $Y_{ij} \sim \text{Bern}(\theta_i\theta_j)$, the expected degree of node i is simply

$$\begin{aligned}
\mathbb{E}[d_i] &= \mathbb{E}\left[\mathbb{E}\left[\sum_{j\neq i} Y_{ij} \middle| \boldsymbol{\theta}\right]\right] \\
&= \mathbb{E}\left[\sum_{j\neq i} \mathbb{E}[Y_{ij}|\boldsymbol{\theta}]\right] \\
&= \mathbb{E}\left[\sum_{j\neq i} \theta_i\theta_j\right] \\
&= (n-1)\mathbb{E}[\theta_i]^2 \\
&= (n-1)\frac{\alpha^2}{(\alpha+\beta)^2}
\end{aligned}$$

since all θ_i are i.i.d.. To compute the variance of d_i, we have:

$$\mathrm{Var}[d_i] = \mathrm{Var}\left[\sum_{j\neq i} Y_{ij}\right] = \sum_{j\neq i}\mathrm{Var}[Y_{ij}] + 2\sum_{\substack{j<k\\ j\neq i\\ k\neq i}}\mathrm{Cov}[Y_{ij}, Y_{ik}].$$

Observe that

$$\mathbb{E}[Y_{ij}] = \mathbb{E}[\mathbb{E}[Y_{ij}|\boldsymbol{\theta}]] = \mathbb{E}[\theta_i\theta_j] = \mathbb{E}[\theta_i]^2 = \left(\frac{\alpha}{\alpha+\beta}\right)^2$$

which by definition of the Bernoulli means we can write $Y_{ij} \sim \mathrm{Bern}\left((\frac{\alpha}{\alpha+\beta})^2\right)$. So

$$\mathrm{Var}[Y_{ij}] = \left(\frac{\alpha}{\alpha+\beta}\right)^2\left(1-\left(\frac{\alpha}{\alpha+\beta}\right)^2\right).$$

For the covariance, we have

$$\begin{aligned}
\mathbb{E}[Y_{ij}Y_{ik}] &= \mathbb{E}[\mathbb{E}[Y_{ij}Y_{ik}|\boldsymbol{\theta}]] \\
&= \mathbb{E}[\theta_i^2\theta_j\theta_k] \\
&= \mathbb{E}[\theta_i^2]\mathbb{E}[\theta_i]^2 \\
&= \frac{\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)}\left(\frac{\alpha}{\alpha+\beta}\right)^2 \\
&= \frac{\alpha^3(\alpha+1)}{(\alpha+\beta)^3(\alpha+\beta+1)}
\end{aligned}$$

and

$$\mathbb{E}[Y_{ij}]\mathbb{E}[Y_{ik}] = \left(\frac{\alpha}{\alpha+\beta}\right)^4.$$

Thus, as $\binom{n-1}{2}$ covariance terms appear in the expression for $\mathrm{Var}[d_i]$,

$$\mathrm{Var}[d_i] = (n-1)\left(\frac{\alpha}{\alpha+\beta}\right)^2\left(1-\left(\frac{\alpha}{\alpha+\beta}\right)^2\right)+(n-1)(n-2)\left[\frac{\alpha^3(\alpha+1)}{(\alpha+\beta)^3(\alpha+\beta+1)}-\left(\frac{\alpha}{\alpha+\beta}\right)^4\right].$$

Going a step further, we can actually compute the entire probability distribution of node degrees (discrete over $\{0,\ldots,n-1\}$) using a nice trick: for fixed i, applying the law of total expectation gives that the distribution of $\sum_{j\neq i} Y_{ij}|\theta_i$ is Binomial$(n-1, \theta_i(\frac{\alpha}{\alpha+\beta}))$. Using this, we obtain:

$$\begin{aligned}
\mathbb{P}(d_i = k) &= \int_0^1 \mathbb{P}\left(\sum_{j\neq i} Y_i = k\,|\theta_i\right) f_{\mathrm{Beta}}(\theta_i;\alpha,\beta)\,d\theta_i \\
&= \int_0^1 \binom{n-1}{k}\left(\frac{\alpha\theta_i}{\alpha+\beta}\right)^k\left(1-\frac{\alpha\theta_i}{\alpha+\beta}\right)^{n-k-1}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta_i^{\alpha-1}(1-\theta_i)^{\beta-1}
\end{aligned}$$

$$= \frac{\Gamma(\alpha+\beta)\binom{n-1}{k}\left(\frac{\alpha}{\alpha+\beta}\right)^k \Gamma(k+\alpha)\,{}_2\tilde{F}_1\left(k-n+1, k+\alpha; k+\alpha+\beta; \frac{\alpha}{\alpha+\beta}\right)}{\Gamma(\alpha)}$$

where f_{Beta} is the PDF for the Beta and ${}_2\tilde{F}_1$ is the *regularized hypergeometric function*, defined by the infinite series

$${}_2\tilde{F}_1(a, b; c; z) = \sum_{j=0}^{\infty} \frac{(a)_j (b)_j z^j}{\Gamma(c+j) j!}$$

where $(x)_j = \frac{\Gamma(x+j)}{\Gamma(j)}$. A key property of ${}_2\tilde{F}_1$, which we will not discuss in detail in this paper, is that if a is a non-positive integer (which is the case here), then ${}_2\tilde{F}_1(a, b; c; z)$ is equal to a polynomial function in z of finite degree (*NIST Digital Library of Mathematical Functions*). This tells us that $\mathbb{P}(d_i = k)$ is expressible in terms of the Gamma function and finite-degree polynomials.

The shape of this discrete distribution will obviously depend on the values of n, α, and β. We were able to use R to plot the discrete probabilities for some select values of the parameters, validating that these distributions are degree-heterogenous. Due to numerical instability we consider relatively small n. For $n = 50$ and $\alpha = \beta = 3$ (so the latent distribution of degree effects resembles a bell curve centered around 0.5), the degree distribution also appears bell-shaped, though right-skewed. This matches with the intuition that edge formation depends multiplicatively on both nodes' degree effects (if the degree effects are distributed i.i.d. with mean 0.5, the probability of edge formation between two nodes should be distributed with mean < 0.5). For $n = 40$ and $\alpha = 3$, $\beta = 1$, the latent distribution has more mass close to 1, so we expect the degree distribution to have more mass at greater values; this is indeed what happens. Figure 1 displays these plots.

Figure 2 considers $n = 100$ and $\alpha = 1$, $\beta = 5$, corresponding to conditions that we believe would be most conducive to creating a scale-free network, in that most degree effects are small except for a few. As can be seen in the figure, the resulting distribution of degrees is heavily right-skewed. However, taking a log-log plot reveals that the distribution does *not* follow a power law (for if it did, we would have $P(d_i = k) \propto k^{-\lambda} \implies \log P(d_i = k) \propto -\lambda \log(k)$ for some λ, producing a linear relationship in the log-log plot). To produce power law distributions, we believe that a latent distribution other than the Beta would be needed. Nevertheless, we believe the plots shown give evidence that our model can capture a range of degree heterogeneity.

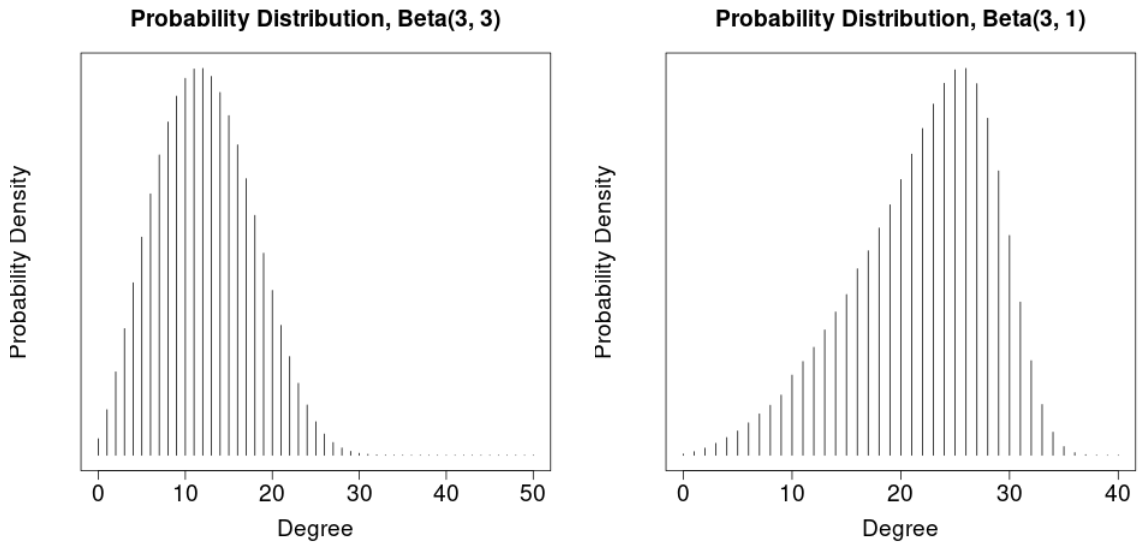

Figure 1: Calculated probability densities for the distribution where $n = 50$ and $\rho = \text{Beta}(3, 3)$ at left, and where $n = 40$ and $\rho = \text{Beta}(3, 1)$ at right.

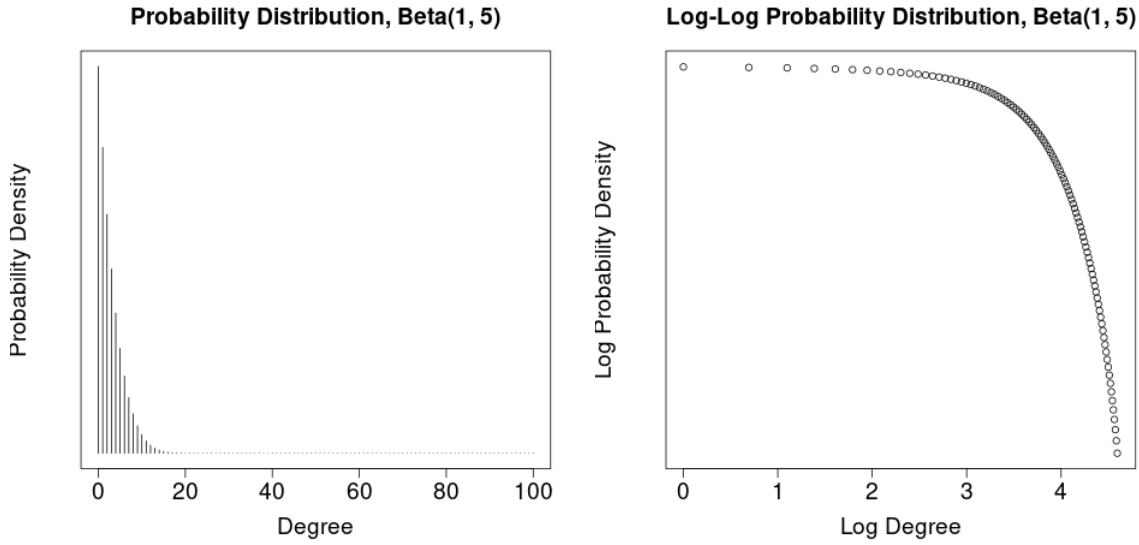

Figure 2: Standard and log-log plot of the probability density for the distribution where $n = 100$ and $\rho = \text{Beta}(1, 5)$.

3.3 Inference using Method of Moments

We consider how to construct estimates for the underlying parameters α and β from a single observation of a graph, in the form of the adjacency matrix y_{ij}. Here we propose a Method of Moments (MoM) approach. A natural alternative choice is to consider maximum likelihood, but our derivations did not lead to tractable solutions; see Appendix A for details.

Recall that y_{ij} is our data (the crystallized Y_{ij}s). The empirical mean, μ, of the y_{ij}'s is:

$$\mu = \frac{1}{\binom{n}{2}} \sum_{i<j} y_{ij}.$$

The empirical variance is not helpful for estimating α and β, as it is a deterministic function of μ. Instead, we consider the empirical estimate of $\mathbb{E}[Y_{ij}Y_{k\ell}]$ over all i, j, k, ℓ such that the sets $\{i, j\}$ and $\{k, \ell\}$ have exactly one element in common; for notation WLOG let it be i. This is equivalent to the number of length-2 paths, or "V-shapes", in the graph, or the number of ways to choose two edges from the same node. The empirical version of this average, which we will call τ, is given by:

$$\tau = \frac{1}{\binom{n}{3}} \sum_{i<j<k} \frac{y_{ij}y_{ik} + y_{ij}y_{jk} + y_{ik}y_{jk}}{3}.$$

When α and β are known, the true values of these quantities are

$$\mathbb{E}[Y_{ij}] = \left(\frac{\alpha}{\alpha+\beta}\right)^2$$

as found above, and

$$\begin{aligned}
\mathbb{E}[Y_{ij}Y_{ik}] &= \mathbb{E}[\mathbb{E}[Y_{ij}Y_{ik}|\boldsymbol{\theta}]] \\
&= \mathbb{E}[\theta_i^2\theta_j\theta_k] \\
&= \mathbb{E}[\theta_i^2]\mathbb{E}[\theta_i]^2 \\
&= \frac{\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)} \left(\frac{\alpha}{\alpha+\beta}\right)^2 \\
&= \frac{\alpha^3(\alpha+1)}{(\alpha+\beta)^3(\alpha+\beta+1)}.
\end{aligned}$$

So, to derive a MoM estimator $(\hat{\alpha}, \hat{\beta})$, we solve the following system of equations:

$$\left(\frac{\hat{\alpha}}{\hat{\alpha}+\hat{\beta}}\right)^2 = \mu := \frac{1}{\binom{n}{2}} \sum_{i<j} y_{ij}$$

$$\frac{\hat{\alpha}^3(\hat{\alpha}+1)}{(\hat{\alpha}+\hat{\beta})^3(\hat{\alpha}+\hat{\beta}+1)} = \tau := \frac{1}{\binom{n}{3}} \sum_{i<j<k} \frac{y_{ij}y_{ik} + y_{ij}y_{jk} + y_{ik}y_{jk}}{3}.$$

Some algebraic manipulation, omitted here for brevity, gives the resulting estimators:

$$\hat{\alpha} = \frac{\tau\sqrt{\mu} \pm \mu^2}{\mu^2 - \tau}$$
$$\hat{\beta} = \frac{-\hat{\alpha}\sqrt{\mu} \mp \hat{\alpha}}{\sqrt{\mu}}$$

As we will see in the next section, this technique is quite effective for obtaining reliable estimates of α and β. However, it faces the possible limitation of returning negative values of $\hat{\alpha}$ and $\hat{\beta}$ in certain circumstances, even when we account for both possible solutions.

4 Simulations

4.1 Model Simulations

Earlier we saw how the derived PDF expresses degree heterogeneity. To extend on this empirically, we generate and visualize some graphs using our method in Figure 3. For sake of visualization, we set $n = 25$. Choosing the three sets of parameters as above ($\alpha = 3, \beta = 1$; $\alpha = 3, \beta = 3$; $\alpha = 1, \beta = 5$) shows that the graphs display drastically different levels of clustering as expected. Notably, for the $\alpha = 1, \beta = 5$ case, the graph contains many entirely-disconnected nodes.

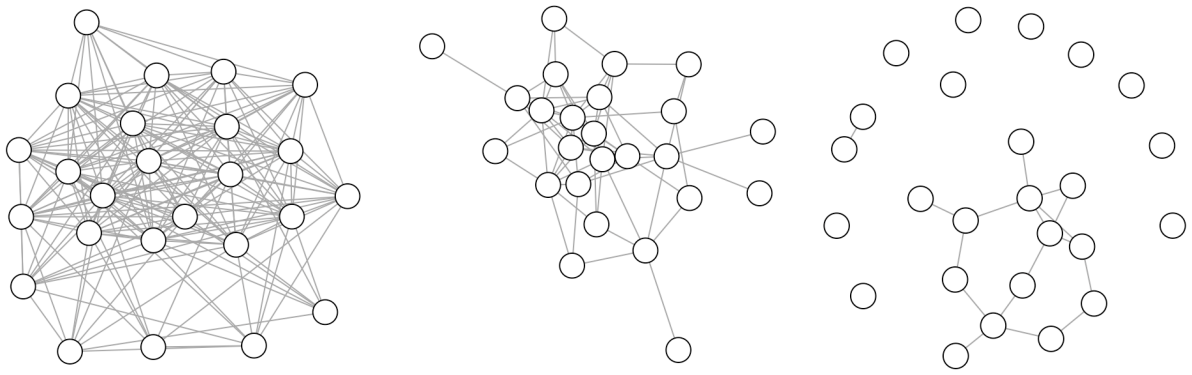

Figure 3: Visualizations of graphs with $n = 25$ and parameters (left to right) $\alpha = 3, \beta = 1$; $\alpha = \beta = 3$; $\alpha = 1, \beta = 5$.

Now setting $n = 1000$, we produce histograms of the empirical distribution of node degrees in generated graphs to validate against our PDF. We generate 100 graphs for each of the three choices

of α and β, and present the aggregate counts over all degrees in Figure 4. As expected, the counts resemble the shape of the probability distributions obtained in Figures 1 and 2, illustrating varied degree heterogeneity with our choice of parameters.

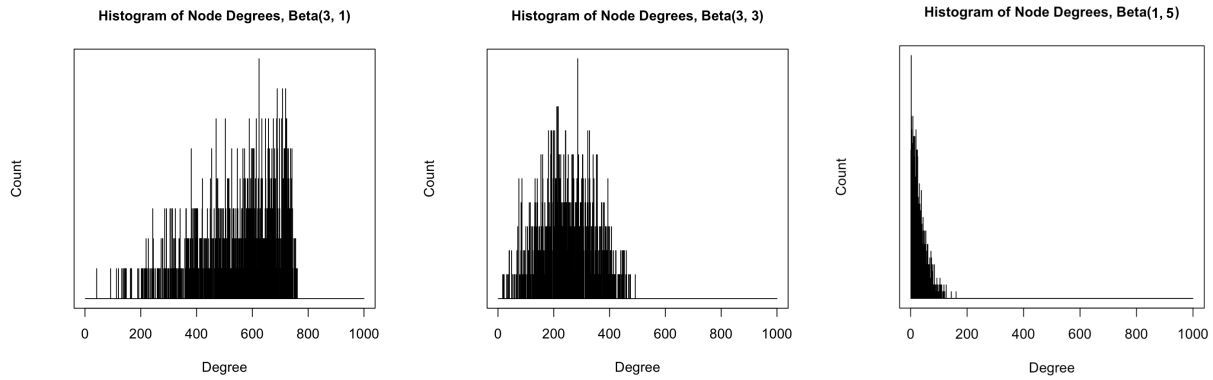

Figure 4: Histograms of aggregate node degree counts over 100 graphs with $n = 1000$ and parameters (left to right) $\alpha = 3, \beta = 1$; $\alpha = \beta = 3$; $\alpha = 1, \beta = 5$.

4.2 Reliability of Method of Moments Estimation

To test our MoM estimators, we simulated 1000 random graphs using our model with $n = 100$, $\alpha = 2$, and $\beta = 5$, and then estimated $\hat{\alpha}$ and $\hat{\beta}$ using our proposed approach. Our estimates for the two parameters are summarized in Figure 5. Across replications, the mean of $\hat{\alpha}$ was 2.21 and the mean of $\hat{\beta}$ was 5.54. We believe the variance in our estimated values is due to the relatively-small value of n, which introduces noise due to the small volume of data available. Notably, however, the histograms show a right-skew, implying that some runs gave inappropriately-high estimated values. A next step for us would be to investigate why this may occur (or if this is just due to noise).

4.3 Application to Real Datasets

We applied our Method of Moments estimattion approach to five common real-world network datasets, provided by Professor Ke: `CoAuthor`, `Dolphin`, `Karate`, `Polbooks`, and `UKFaculty`. The estimates of $\hat{\alpha}$ and $\hat{\beta}$ we obtained are provided in Table 1. We find that the estimates have a very high ratio $\frac{\hat{\beta}}{\hat{\alpha}}$, suggesting sparsity in the graph (larger ratios mean the Beta distribution has more mass closer to 0, implying that nodes do not have as many neighbors).

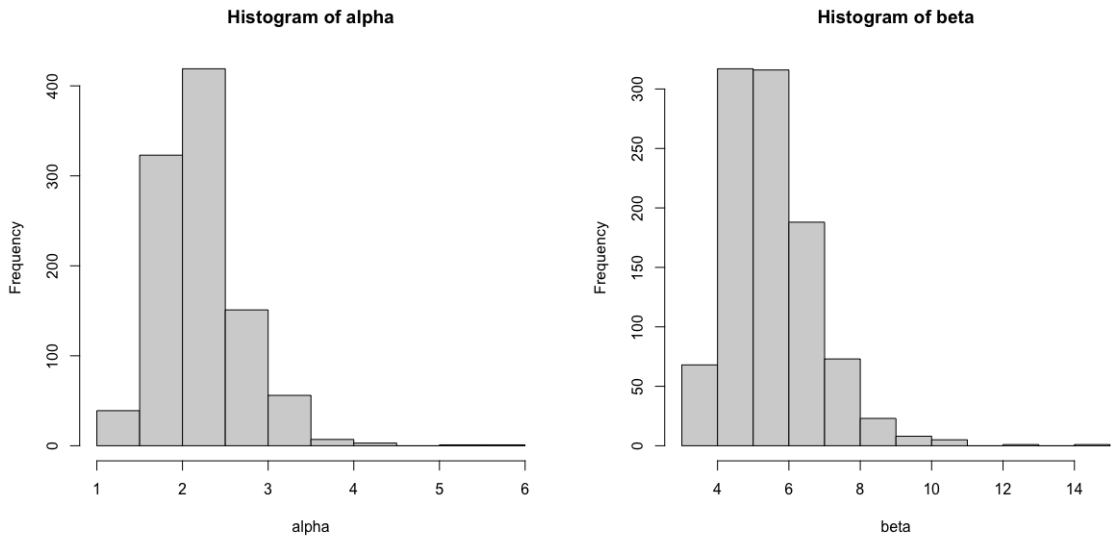

Figure 5: Histogram of estimated values of $\hat{\alpha}$ (left) and $\hat{\beta}$ (right) across 1000 generated graphs with $n = 100, \alpha = 2, \beta = 5$.

Dataset	$\hat{\alpha}$	$\hat{\beta}$
`CoAuthor`	1.35	86.8
`Dolphin`	4.42	24.5
`Karate`	0.830	16.8
`Polbooks`	1.99	25.2
`UKFaculty`	1.72	13.7

Table 1: Estimates of α and β from our model derived from provided real-world datasets.

5 Discussion and Conclusion

In this report we have proposed a simple hierarchical model meant to capture degree heterogeneity in graphs. Omitting the idea of communities or groups, we first assign each node a degree effect parameter drawn from an underlying distribution ρ over $[0, 1]$, and then generate edges between nodes with probability proportional to the two nodes' degree effects. Here we considered the case where ρ is the Beta distribution. We saw that choosing different values for the parameters of Beta allowed us to express various different distributions in node degrees, and that using the method of moments allowed us to produce estimators for the parameters from observed data.

The work presented immediately raises a few questions that can be tackled as follow-up work.

We saw that no set of Beta parameters tested with our model produced a power law distribution in node degrees, a feature of scale-free networks that commonly appear in the real world. It would be interesting to investigate if there exist other distributions over $[0, 1]$ with this model setup that produce the desired power-law distributions. In general, while our report only considers the Beta, we believe it could be fascinating to consider the behavior of graphs when ρ is some other latent distributions over $[0, 1]$. Even more generally, it would be interesting to consider latent distributions with arbitrary support, which can work so long as we "clip" the probabilities of edge generation themselves to lie in $[0, 1]$.

To construct our model we omitted the consideration of communities, so another natural extension is to reintroduce community effects. Indeed, consider the generalized model that partitions the nodes into communities $\mathcal{C}_1, \ldots, \mathcal{C}_K$, contains a (symmetric) matrix of community effects $\mathbf{P} = (P_{k\ell})$, and sets the probability for edge generation for $P(Y_{ij} = 1) = \theta_i\theta_j + P_{k\ell}$ where $\theta_i, \theta_j \sim \rho$ i.i.d., $i \in \mathcal{C}_k$, and $j \in \mathcal{C}_\ell$. ρ itself could even vary among communities. This could be considered a modification of DCBM imposing structure on degree effects. However, we are *a priori* uncertain about how feasibly this kind of model would lend itself to performing statistical inference. Bayesian methods in which we impose a prior on the degree effect distribution might be a possible pathway for inference.

There is also much more that can theoretically be understood about the model we described, even in the case where ρ is the Beta. As part of our Method of Moments estimation, we remarked above that we calculate the number of "V-shapes" in the graph. (Jin, Ke, and Luo 2018) showed that in the DCBM model, counting the number of such short paths and cycles prove useful in testing for the presence of communities. It may be interesting to study whether or not properties of the model and ρ can be deduced from the distributions of these more general shapes. Additionally, we have not presented any work discussing whether or not our estimators are consistent or asymptotically Normal; further analysis of these estimators to examine their behavior could give theoretical backing to the experimental results we found above.

Bibliography

Albert, Réka, and Albert-László Barabási. 2002. "Statistical mechanics of complex networks." *Reviews of Modern Physics* 74, no. 1 (January): 47–97. https://doi.org/10.1103/revmodphys.74.47. http://dx.doi.org/10.1103/RevModPhys.74.47.

Barabási, Albert-László, and Réka Albert. 1999. "Emergence of Scaling in Random Networks." *Science* 286 (5439): 509–512. https://doi.org/10.1126/science.286.5439.509. eprint: https://www.science.org/doi/pdf/10.1126/science.286.5439.509. https://www.science.org/doi/abs/10.1126/science.286.5439.509.

Barabási, G. Bianconi A. -L. 2000. *Competition and multiscaling in evolving networks.* arXiv: cond-mat/0011029 [cond-mat.dis-nn].

Barrat, A., and M. Weigt. 1999. *On the properties of small-world network models.* arXiv: cond-mat/9903411 [cond-mat.dis-nn].

NIST Digital Library of Mathematical Functions. https://dlmf.nist.gov/, Release 1.1.11 of 2023-09-15. F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller, B. V. Saunders, H. S. Cohl, and M. A. McClain, eds. https://dlmf.nist.gov/.

Jin, Jiashun. 2015. "Fast community detection by SCORE." *The Annals of Statistics* 43, no. 1 (February). https://doi.org/10.1214/14-aos1265. http://dx.doi.org/10.1214/14-AOS1265.

Jin, Jiashun, Zheng Tracy Ke, and Shengming Luo. 2018. *Network Global Testing by Counting Graphlets.* arXiv: 1807.08440 [stat.ME].

Karrer, Brian, and M. E. J. Newman. 2011. "Stochastic blockmodels and community structure in networks." *Physical Review E* 83, no. 1 (January). https://doi.org/10.1103/physreve.83.016107. http://dx.doi.org/10.1103/PhysRevE.83.016107.

Lee, Clement, and Darren J. Wilkinson. 2019. "A review of stochastic block models and extensions for graph clustering." *Applied Network Science* 4 (1): 122. https://doi.org/10.1007/s41109-019-0232-2. https://doi.org/10.1007/s41109-019-0232-2.

Lu, Xiaoyan, and Boleslaw K. Szymanski. 2019. "A Regularized Stochastic Block Model for the robust community detection in complex networks." *Scientific Reports* 9 (1): 13247. https://doi.org/10.1038/s41598-019-49580-5. https://doi.org/10.1038/s41598-019-49580-5.

Noh, Jae Dong. 2003. "Exact scaling properties of a hierarchical network model." *Physical Review E* 67, no. 4 (April). https://doi.org/10.1103/physreve.67.045103. http://dx.doi.org/10.1103/PhysRevE.67.045103.

Ravasz, Erzsébet, and Albert-László Barabási. 2003. "Hierarchical organization in complex networks." *Physical Review E* 67, no. 2 (February). https://doi.org/10.1103/physreve.67.026112. http://dx.doi.org/10.1103/PhysRevE.67.026112.

Servedio, Vito D. P., Guido Caldarelli, and Paolo Buttà. 2004. "Vertex intrinsic fitness: How to produce arbitrary scale-free networks." *Physical Review E* 70, no. 5 (November). https://doi.org/10.1103/physreve.70.056126. http://dx.doi.org/10.1103/PhysRevE.70.056126.

Watts, Duncan J., and Steven H. Strogatz. 1998. "Collective dynamics of 'small-world'networks." *Nature* 393 (6684): 440–442. https://doi.org/10.1038/30918. https://doi.org/10.1038/30918.

Appendix A Maximum Likelihood Estimation

UPDATE: This analysis isn't quite correct, because the likelihood is un-normalized. To properly use E-M, we would need to compute the expectation of the *normalized* log-likelihood.

A natural first idea in estimating α and β given data is to use maximum likelihood inference. Let $\mathbf{Y}$ denote the (symmetric) adjacency matrix of our observed graph. The likelihood function for α and β is given by:

$$\begin{aligned}
L(\alpha, \beta; \mathbf{Y}) &= \pi(\mathbf{Y}|\alpha, \beta) \\
&= \int\limits_{\boldsymbol{\theta} \in [0,1]^n} \pi(\mathbf{Y}, \boldsymbol{\theta}|\alpha, \beta)\, d\boldsymbol{\theta} \\
&= \int\limits_{\boldsymbol{\theta} \in [0,1]^n} \pi(\mathbf{Y}|\boldsymbol{\theta}, \alpha, \beta)\pi(\boldsymbol{\theta}|\alpha, \beta)\, d\boldsymbol{\theta} \\
&= \int\limits_{\boldsymbol{\theta} \in [0,1]^n} \left(\prod_{i<j} (\theta_i \theta_j)^{y_{ij}} (1 - \theta_i \theta_j)^{1 - y_{ij}} \right) \left(\prod_{i \leq n} \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \theta_i^{\alpha - 1} (1 - \theta_i)^{\beta - 1} \right) d\boldsymbol{\theta},
\end{aligned}$$

which is a n-dimensional integral. To the extent of our knowledge, this integral is intractable, but the E-M algorithm provides a way to work with this likelihood. To implement this algorithm, we compute the expectation of the complete-data log-likelihood:

$$\begin{aligned}
\mathbb{E}[\log L(\alpha, \beta; \mathbf{Y}, \boldsymbol{\theta})] &= \mathbb{E}\left[\log \left[\left(\prod_{i<j} (\theta_i \theta_j)^{y_{ij}} (1 - \theta_i \theta_j)^{1 - y_{ij}} \right) \left(\prod_{i \leq n} \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \theta_i^{\alpha - 1} (1 - \theta_i)^{\beta - 1} \right) \right] \right] \\
&= \mathbb{E}\Bigg[\sum_{i<j} y_{ij} \log(\theta_i \theta_j) + \sum_{i<j} (1 - y_{ij}) \log(1 - \theta_i \theta_j) \\
&\qquad + (\alpha - 1) \sum_{i \leq n} \log(\theta_i) + (\beta - 1) \sum_{i \leq n} \log(1 - \theta_i) \\
&\qquad + n(\log \Gamma(\alpha + \beta) - \log \Gamma(\alpha) - \log \Gamma(\beta)) \Bigg]
\end{aligned}$$

Let $S = \sum_{i<j} y_{ij}$ be the sufficient statistic, and let $T = \sum_{i<j} (1 - y_{ij}) = \binom{n}{2} - S$. We compute that

$$\begin{aligned}
\mathbb{E}[\log \theta_i] &= \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \int_0^1 \log(x) x^{\alpha - 1} (1 - x)^{\beta - 1}\, dx = \psi(\alpha) - \psi(\alpha + \beta) \\
\mathbb{E}[\log(1 - \theta_i)] &= \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \int_0^1 \log(1 - x) x^{\alpha - 1} (1 - x)^{\beta - 1}\, dx = \psi(\beta) - \psi(\alpha + \beta) \\
\mathbb{E}[\log(\theta_i \theta_j)] &= \mathbb{E}[\log \theta_i] + \mathbb{E}[\log \theta_j] = 2(\psi(\alpha) - \psi(\alpha + \beta))
\end{aligned}$$

where ψ is the digamma function. To the extent of our knowledge, $\mathbb{E}[\log(1-\theta_i\theta_j)]$ does not have a closed-form solution, but for small values of θ_i (i.e. the matrix is sparse), it may be approximated by the following first-order Taylor expansion:

$$\mathbb{E}[\log(1-\theta_i\theta_j)] \approx \mathbb{E}[-\theta_i\theta_j] = -\left(\frac{\alpha}{\alpha+\beta}\right)^2.$$

Using this, the approximate expectation of the log-likelihood is given by

$$\begin{aligned}\mathbb{E}[\log L(\alpha,\beta;\mathbf{Y},\boldsymbol{\theta})] \approx{}& 2S(\psi(\alpha)-\psi(\alpha+\beta)) - T\left(\frac{\alpha}{\alpha+\beta}\right)^2 \\ &+ n(\alpha-1)(\psi(\alpha)-\psi(\alpha+\beta)) + n(\beta-1)(\psi(\beta)-\psi(\alpha+\beta)) \\ &+ n(\log\Gamma(\alpha+\beta)-\log\Gamma(\alpha)-\log\Gamma(\beta)).\end{aligned}$$

Its approximate gradient has a simple closed form:

$$\nabla\mathbb{E}[\log L(\alpha,\beta;\mathbf{Y},\boldsymbol{\theta})] \approx \begin{bmatrix} -\psi^{(1)}(\alpha+\beta)(n(\alpha+\beta-2)+2S) + \psi^{(1)}(\alpha)((\alpha-1)n+2S) - \frac{2\alpha\beta T}{(\alpha+\beta)^3} \\ (\beta-1)n\psi^{(1)}(\beta) - \psi^{(1)}(\alpha+\beta)(n(\alpha+\beta-2)+2S) + \frac{2\alpha^2 T}{(\alpha+\beta)^3} \end{bmatrix}$$

where $\psi^{(1)}$ is the trigamma function, equal to the derivative of the digamma function. So, a sketch of the EM algorithm to find the MLE $(\hat{\alpha},\hat{\beta})$ is as follows:

1. Initialize values $0<\alpha_0<1$, $0<\beta_0<1$, step size $\epsilon>0$, tolerance $\tau>0$, and number of iterations $N>0$.

2. For $i>0$, iteratively compute $\nabla\mathbb{E}[\log L(\alpha_{i-1},\beta_{i-1};\mathbf{Y},\boldsymbol{\theta})]$, and set

$$\begin{bmatrix}\alpha_i \\ \beta_i\end{bmatrix} = \begin{bmatrix}\alpha_{i-1} \\ \beta_{i-1}\end{bmatrix} + \epsilon\nabla\mathbb{E}[\log L(\alpha_{i-1},\beta_{i-1};\mathbf{Y},\boldsymbol{\theta})].$$

3. Stop when $i=N$, or when $\max(|\alpha_i-\alpha_{i-1}|,|\beta_i-\beta_{i-1}|)<\tau$, whichever happens first. Return $\hat{\alpha}=\alpha_i$ and $\hat{\beta}=\beta_i$.

Confusingly, when implementing this algorithm through experimentation, we observed *no* convergence to a MLE value MLE. Instead, the estimates for α and β would diverge to $+\infty$ while maintaining a constant ratio $\frac{\alpha}{\alpha+\beta}=\sqrt{\frac{S}{\binom{n}{2}}}$. Intuitively, this appears to be due to a kind of overfitting: if each θ_i were deterministically set to equal $\sqrt{\frac{S}{\binom{n}{2}}}$, we can choose α and β such that the Beta(α,β)

distribution approximates a Dirac-Delta function centered at θ_i. In this case, the (log-)likelihood will diverge to $+\infty$, as α and β can get arbitrarily large while still increasing the likelihood.

Indeed, we ran the E-M algorithm described in the previous section on a random graph generated using $\alpha = 2$ and $\beta = 7$. We set $\epsilon = 0.001$ and ran the algorithm at 10^7 iterations, reporting parameter values ever 1000 iterations. As shown in Figure 6, the parameters diverged to infinity while maintaining a constant ratio.

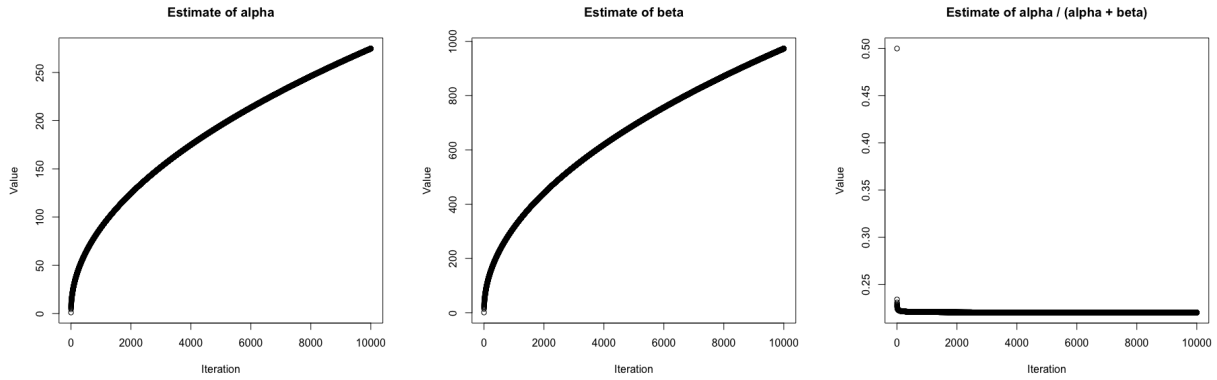

Figure 6: Estimated values of α_i (left), β_i (center),and the ratio $\frac{\alpha_i}{\alpha_i+\beta_i}$ (right) over successive iterations of the E-M algorithm.

Appendix B Data and Code

Real-world datasets were provided by Prof. Tracy Ke with permission. Our code was written entirely in R, and is available online at `https://github.com/ispecht/stat-236/tree/main`.