Chapter 6
Models for Network Graphs

In this chapter, we turn to the topic of modeling network graphs. We introduce a number of important classes of network graph models, and we illustrate some of the various statistical uses to which these models have been put, including the detection of network motifs, the evaluation of proposed network generative mechanisms, and the assessment of potential predictive factors of relational ties.

6.1 Introduction

So far in this book, the emphasis has been almost entirely focused upon methods, to the exclusion of modeling – methods for constructing network graphs, for analyzing their observed structure, and for obtaining sample-based estimates of unobserved or partially observed structure. For the remainder of this book, our focus will shift to the construction and use of models in the analysis of network data. And we begin, in this chapter, with the modeling of network graphs.

By a model for a network graph we mean effectively a collection

$$\{\mathbb{P}_\theta(G), G \in \mathcal{G} : \theta \in \Theta\} \ , \tag{6.1}$$

where $\mathcal{G}$ is a collection (or 'ensemble') of possible graphs, $\mathbb{P}_\theta$ is a probability distribution on $\mathcal{G}$, and θ is a vector of parameters, ranging over possible values in Θ. When convenient, we may often drop the explicit reference to θ and simply write $\mathbb{P}$ for the probability function.[1]

The richness of network graph modeling derives largely from how we choose to specify $\mathbb{P}(\cdot)$, with methods in the literature ranging from the simple to the complex. Some approaches, for example, simply let $\mathbb{P}(\cdot)$ be uniform on $\mathcal{G}$, but restrict $\mathcal{G}$ to contain only those graphs G satisfying certain properties of interest. Other approaches induce $\mathbb{P}(\cdot)$ implicitly through the recurrent application of simple gen-

[1] More specifically, by a 'model' like that in (6.1) we have in mind the specification of a collection of probability spaces, indexed by θ; but we will not work at that level of formality here.

E.D. Kolaczyk, *Statistical Analysis of Network Data*, Springer Series in Statistics,
DOI 10.1007/978-0-387-88146-1_6, © Springer Science+Business Media, LLC 2009

erative mechanisms. Finally, still other approaches are designed to facilitate the inclusion of various endogenous and exogenous factors in the model and the statistical evaluation of their respective association with the topology of a graph G.

In practice, network graph models are used for a variety of purposes, many of which have been alluded to already in earlier chapters. These include the study of proposed mechanisms for the emergence of certain commonly observed properties in real-world networks (such as broad degree distributions or small-world effects) or the testing for 'significance' of a pre-defined characteristic(s) in a given network graph. Also, the use of network graph models allows for an alternative context to the design-based paradigm introduced in the previous chapter for defining and evaluating the accuracy of estimators of summaries of graph structure.

We will discuss four categories of network graph models in this chapter. First, in Section 6.2, we consider random graph models and several of their statistical applications. These models most literally capture the idea of a graph being drawn 'at random' from a collection $\mathscr{G}$. Second, in Section 6.3, we will present models for graphs possessing the small-world property observed in many real-world network graphs, as described in Section 4.3.2. Third, in Section 6.4, we will see a number of network growth models, wherein the focus is on the effect of simple, parameterized generative mechanisms. Fourth, and lastly, in Section 6.5, we turn our attention to exponential random graph models, which extend the idea of a statistical regression model to random graphs. A case study that examines the application of exponential random graph models to patterns of cooperation among corporate lawyers will be presented in Section 6.5.4. Some additional comments on the significant challenges remaining for network graph modeling are described briefly in Section 6.6.

6.2 Random Graph Models

The term *random graph model* typically is used to refer to a model specifying a collection $\mathscr{G}$ and a uniform probability $\mathbb{P}(\cdot)$ over $\mathscr{G}$. Random graph models are arguably the most well-developed class of network graph models, mathematically speaking. This is no doubt due in part to the comparatively simpler nature of these models. However, it is precisely this nature that has allowed for, in contrast to many of the other classes of network graph models, the precise analytical characterization of many of the structural summary measures introduced in Chapter 4. And these characterizations in turn help provide insight into appropriate usage of such models for estimation and testing of structural characteristics in observed network graphs.

Example 6.1 (Model-Based Estimation in Network Graphs). In traditional statistical sampling theory, there are two main approaches to constructing estimates of population parameters from a sample: design-based and model-based. In the design-based approach, inference is based entirely on the random mechanism by which a subset of elements were selected from the population to create the sample. In the model-based approach, on the other hand, a model is given by the analyst that

specifies a relationship between the sample and the population. Model-based estimation strategies (e.g., least-squares, method-of-moments, maximum-likelihood, etc.) are then typically used for constructing estimators of population parameters. In more recent decades, the distinction between these two approaches has become more blurred, with elements of both design- and model-based strategies often used profitably in conjunction with each other. See the book by Särndal, Swensson, and Wretman [339], for example.

Consider the task of estimating a given characteristic $\eta(G)$ of a network graph G, based on a sampled version of that graph, G^*. In Chapter 5, we presented almost exclusively a design-based perspective on this problem. Alternatively, we might augment this perspective to included a model-based component as well, wherein G is assumed to have been generated uniformly at random from a collection $\mathscr{G}$, prior to our obtaining G^*. Inference on η then should incorporate both randomness due to selection of G from $\mathscr{G}$ and randomness due to sampling G^* from G. We will illustrate this approach for estimating the size of a 'hidden' population, using one of the models introduced in this section, later in Section 6.2.4. □

Example 6.2 (Assessing Significance in Network Graphs). Suppose that we have a graph derived from observations of some sort (i.e., not necessarily through a formal network sampling mechanism), which we will denote as G^{obs} here, and that we are interested in some structural characteristic, say $\eta(\cdot)$. In particular, suppose that we are interested in assessing whether the value $\eta(G^{obs})$ is 'significant,' in the sense of being somehow unusual or unexpected.

Of course, such notions of significance must be defined in comparison to an appropriate frame of reference. Random graph models are often used in setting up such comparisons. A collection of random graphs $\mathscr{G}$ is defined, and the value $\eta(G^{obs})$ is compared to the collection of values $\{\eta(G) : G \in \mathscr{G}\}$. If $\eta(G^{obs})$ is judged to be extreme with respect to this collection, then that is taken as evidence that G^{obs} is unusual in having this value. More formally, a random graph model is used to create a reference distribution which, under the accompanying assumption of uniform likelihood of elements in $\mathscr{G}$, takes the form

$$\mathbb{P}_{\eta,\mathscr{G}}(t) = \frac{\#\{G \in \mathscr{G} : \eta(G) \leq t\}}{|\mathscr{G}|} \quad . \tag{6.2}$$

If $\eta(G^{obs})$ is found to be sufficiently unlikely under this distribution, this is taken as evidence against the hypothesis that G^{obs} is a uniform draw from $\mathscr{G}$.

How best to choose $\mathscr{G}$ is a practical issue of some importance, since it can have a direct impact on the relevance of the results of such procedures, as we illustrate later in Section 6.2.4. Also important is the fact that, outside of the contexts with both very small network graphs and very simple random graph models $\mathscr{G}$, we cannot generally hope to be able to explicitly enumerate all of the elements of $\mathscr{G}$, and therefore, we cannot expect to be able to calculate the probabilities in (6.2) exactly. Rather, approximations to these probabilities are used. There are analytical approximations for certain comparatively simple cases; see Wasserman and Faust [393, Ch.

13.5], for example. But for most other cases of interest it is necessary to employ Monte Carlo simulation methods – such as those we describe in Section 6.2.3 – to obtain numerical approximations. □

In this section, we present classical random graph models and generalized random graph models, including an overview of various of the most relevant properties of each. We then describe algorithms for simulating random draws of such graphs from their underlying collections $\mathscr{G}$, which is critical for their use in applications. Finally, we illustrate the practical application of random graph models in the contexts of model-based estimation and assessment of significance, as introduced in Examples 6.1 and 6.2. Our focus here will be primarily on graphs G that are undirected and simple (i.e., no loops or multi-edges). Extensions of the ideas and results we discuss to, for example, directed graphs are generally straightforward and may be found in many of the references cited.

6.2.1 Classical Random Graph Models

The classical theory of random graph models, as established in a series of seminal papers by Erdös and Rényi [134, 135, 136], rests upon a simple model that places equal probability on all graphs of a given order and size. Specifically, their model specifies a collection $\mathscr{G}_{N_v,N_e}$ of all graphs $G = (V,E)$ with $|V| = N_v$ and $|E| = N_e$, and assigns probability $\mathbb{P}(G) = \binom{N}{N_e}^{-1}$ to each $G \in \mathscr{G}_{N_v,N_e}$, where $N = \binom{N_v}{2}$ is the total number of distinct vertex pairs.[2] The key contribution of Erdös and Rényi was to develop a foundation of formal probabilistic results concerning the characteristics of graphs G drawn randomly from $\mathscr{G}_{N_v,N_e}$ with respect to this $\mathbb{P}(\cdot)$.

It is more common, however, to see such results stated for a variant of $\mathscr{G}_{N_v,N_e}$, first suggested by Gilbert [173] at approximately the same time. In this formulation, a collection $\mathscr{G}_{N_v,p}$ is defined to consist of all graphs G of order N_v that may be obtained by assigning an edge independently to each pair of distinct vertices with probability $p \in (0,1)$. When p is an appropriately defined function of N_v, and $N_e \sim pN_v$, these two classes of models are essentially equivalent for large N_v. We shall refer primarily to the version of Gilbert when discussing 'classical random graphs' in the remainder of this section.

Classical random graph models are by far the most thoroughly studied in the literature. We mention just a handful of key properties here – primarily those associated with certain of the characteristics mentioned in Chapter 4. A thorough coverage of results of this nature may be found in the book by Bollobás [43].

Note that we cannot expect that a random graph be connected *per se.* However, it is possible to make precise statements regarding the level of connectivity likely

[2] Following convention in the literature, we will frequently abuse notation by annotating our collection $\mathscr{G}$ with additional parameters associated with the probability distribution $\mathbb{P}$ accompanying $\mathscr{G}$, as in $\mathscr{G}_{N_v,N_e}$, and refer to that interchangeably as the 'collection' or the 'model,' as context demands, rather than writing out the full model specification corresponding to (6.1).

in a randomly generated $G \in \mathscr{G}_{N_v,p}$, as a function of the relation between N_v and p. For example, let $p = c/N_v$, for $c > 0$. Then, if $c > 1$, with high probability[3] G will have a single connected component consisting of $\alpha_c N_v$ vertices, for some constant $\alpha_c > 0$ depending on c, with the remaining components having only on the order of $O(\log N_v)$ vertices. On the other hand, if $c < 1$, then all components will have on the order of $O(\log N_v)$ vertices, with high probability – that is, G will consist entirely of a large number of very small, separate components. In the case of $c > 1$, the larger component is typically called the *giant component*, as was mentioned in Section 4.3.2. Note too that, under the stated conditions, the expected density of the graph G is only $p \sim N_v^{-1}$, and so G is likely to be sparse for large N_v.

Regarding the degree distribution, it is known that classical random graphs have distributions that are concentrated, with exponentially decaying tails. More precisely, if we write $f_d(G)$ as the (random) proportion of vertices with degree d in G, then for $d = 0, 1, 2, \ldots$ and $p = c/N_v$, with $c > 0$, it can be shown that with high probability

$$(1-\varepsilon)\frac{c^d e^{-c}}{d!} \leq f_d(G) \leq (1+\varepsilon)\frac{c^d e^{-c}}{d!} \quad , \tag{6.3}$$

for any given $\varepsilon > 0$. This result states that, for large N_v, G will have a degree distribution that is like a Poisson distribution with mean c. That this should be true is somewhat intuitive, as it is easy to see that each vertex $v \in V$ has its degree d_v identically distributed as a binomial random variable,[4] with parameters $N_v - 1$ and p, which suggests the relevance of a Poisson approximation to this binomial with parameter $\mathbb{E}[d_v] = p(N_v - 1) \sim c$.

So classical random graphs do not have the broad degree distribution observed in many large-scale real-world networks. Nor, it turns out, do they display much clustering, under the conditions corresponding to the results above. For example, recall that the clustering coefficient $\mathsf{cl}_T(G)$ measures the fraction of transitive triples in G that close to form a triangle, which may be interpreted as the probability that two neighbors of a randomly chosen vertex v in G are themselves linked. But by construction, this probability is just p, and as we have seen above, the interesting cases involve $p \sim N_v^{-1}$, which will tend to zero as N_v grows.

On the the other hand, graphs in $\mathscr{G}_{N_v,p}$ do possess the small-world property, under the conditions above, in that the diameter can be shown to vary like $O(\log N_v)$, with high probability, as $N_v \to \infty$.

[3] For this result, and a number of others we summarize in this section, variants may be stated with assurances stronger than 'with high probability,' but at the cost of additional technical details and structure.

[4] And in fact, it may be shown that these binomials are quite close to being independent. See McKay and Wormald [274].

6.2.2 Generalized Random Graph Models

The formulation of Erdös and Rényi can be generalized in a straightforward manner. Specifically, the basic recipe is to (a) define a collection of graphs $\mathcal{G}$ consisting of all graphs of a fixed order N_v that possess a given characteristic(s), and then (b) assign equal probability to each of the graphs $G \in \mathcal{G}$. In the Erdös-Rényi model, for example, the condition is simply that the size of the graphs G be equal to some fixed N_e.

Beyond Erdös-Rényi, the most commonly chosen condition is that of a fixed degree sequence. That is, $\mathcal{G}$ is defined to be the collection of all graphs G with a pre-specified degree sequence, which we will write here as $\{d_{(1)}, \ldots, d_{(N_v)}\}$, in ordered form. Note that, with N_v specified as well, the number of edges N_e is fixed under this model, due to the relation $\bar{d} = 2N_e/N_v$. It is also useful to note that, conversely, for fixed N_v and N_e, and a chosen degree sequence with average degree $\bar{d}$ matching this choice, the resulting collection $\mathcal{G}$ is strictly contained within $\mathcal{G}_{N_v,N_e}$. So the addition of an assumed form for the degree sequence is in this case equivalent to specifying our model through a conditional distribution on the original collection $\mathcal{G}_{N_v,N_e}$. In either case, all other characteristics are free to vary to the extent allowed by the chosen degree sequence.

A number of authors have conducted mathematical studies of the properties of generalized random graph models with fixed degree sequence. There are minor differences in the manner in which these authors specify their random graph models, with different formulations better facilitating the use of different mathematical tools. Nonetheless, all of these models are essentially equivalent asymptotically.

Molloy and Reed [282, 283] study random graphs whose degree sequence conforms approximately to a pre-specified degree distribution $\{f_d\}$ and is sparse and well behaved in a certain well-defined sense. They provide a simple condition, on a function of the first two moments of the degree distribution, under which their generalized random graphs G have a giant component, with high probability. If the condition is not satisfied, then, analogous to the classical case, the graph consists of a large collection of small, separate subgraphs. In addition, they show that the giant component, when it exists, should contain a number of vertices growing like αN_v, for fixed constant α, with high probability, similar to the classical case.

Aiello, Chung, and Lu [5] have built upon these results, under a similar model formulation, in the special case where the degree distribution $\{f_d\}$ has a power-law form (i.e., $f_d = Cd^{-\alpha}$). In particular, they provide specific conditions for the emergence of a giant component and they characterize the size of that component and the remaining components, as a function of the parameters α and C. Additionally, Chung and Lu [87] have characterized the diameter and the average distance in a related power-law random graph model, with an eye towards the phenomena of small-world behavior. They have shown, for example, that for degree distribution exponents $\alpha \in (2,3)$, a range cited often in practice,[5] the diameter is of $O(\log N_v)$

[5] Recall, for example, our analysis of the Internet router network in Example 4.2, and the evidence for an exponent α slightly under 3.0.

and the average distance is of order $O(\log\log N_v)$, with high probability under mild conditions.

Finally, using a slightly different modeling framework, Newman, Strogatz, and Watts [299] analyze much the same characteristics of generalized random graph models, using semi-formal arguments, and produce similar results. Later, in the same context, Newman [296, Sec. IV.B] argues that, under certain conditions, for power-law degree distributions with $\alpha \in (7/3, 3)$, the clustering coefficient cl_T should behave like $N_v^{-\beta}$, where $\beta = (3\alpha - 7)/(\alpha - 1)$. This result suggests that, for power-law generalized random graphs with exponent α in this range, the clustering coefficient tends to zero with increasing N_v, but slower than the rate N_v^{-1} observed in the classical setting.

6.2.3 Simulating Random Graph Models

The mathematical study of random graph models yields various fundamental insights. However, in order to use these models in statistical applications it is generally necessary to be able to efficiently simulate the sampling of such random graphs using a computer. For some models it is actually possible to produce samples in linear time; for others, it appears that Markov chain Monte Carlo (MCMC) methods are the only realistic alternative.

6.2.3.1 Simulating Classical Random Graphs

Consider the problem of simulating classical random graph models. In the model $\mathscr{G}_{N_v,p}$, for example, the definition of the model class itself suggests a simple algorithm. That is, in principle, we can simply generate $\binom{N_v}{2} \sim N_v^2/2$ independent Bernoulli random variables, each with probability of success p, and assign edges to those pairs of vertices whose corresponding variable yields a success. Recall, however, that when $p \sim N_v^{-1}$, the graphs emerging from $\mathscr{G}_{N_v,p}$ are expected to be sparse. Therefore, an overwhelming majority of these variables will not yield a success, and the algorithm is inefficient.

Batagelj and Brandes [23], exploiting this fact, offer an algorithm that runs in $O(N_v + N_e)$ time, which in the case of sparse graphs will be significantly faster than the nominal $O(N_v^2)$. The key insight underlying their algorithm is that it is possible to simply skip potential edges in a manner that is nevertheless consistent with the model. In particular, for each vertex $v \in V$, the other $N_v - 1$ vertices are ordered, and an index, say j, is drawn according to a geometric random variable[6] with parameter p. The first $j-1$ potential edges incident to this vertex are not formed, but the j-th one is. If $j < N_v - 1$, so that there remain potential edges, the process is repeated

[6] The waiting time for a success in a sequence of independent and identically distributed Bernoulli(p) random variables is distributed as a geometric random variable with parameter p.

with those that remain. A formal statement of the algorithm is provided in Batagelj and Brandes [23].

For the model $\mathscr{G}_{N_v,N_e}$, recall that the goal is to sample graphs G uniformly from all those with N_v vertices and N_e edges. Batagelj and Brandes [23] observe that, while skipping can again be used, it will be more cumbersome and costly to do so, since the presence or absence of edges are now correlated with each other. More efficient is to simply draw candidate edges one by one, uniformly at random from the set of all $\binom{N_v}{2}$ possible edges, rejecting edges already chosen, until the pre-specified number of distinct edges N_e have been obtained. This algorithm has a running time that is $O(N_v + N_e)$ in expectation.

6.2.3.2 Simulating Generalized Random Graphs

Now let us consider sampling according to a generalized random graph model, where the problem is more challenging than in the classical setting, due to the increased complexity of the specified constraint(s). To illustrate, we will focus our discussion here upon the case in which the collection $\mathscr{G}$ is determined by constraining the degree sequence to be fixed. We offer a few brief comments regarding more general cases at the end of this section.

Two standard approaches to simulating random graphs with fixed degree sequence are *matching algorithms* and *switching algorithms*. The basic matching algorithm is simple and can be described as follows. Given an ordered degree sequence $\{d_{(1)},\ldots,d_{(N_v)}\}$, create a list containing, for each $i = 1,\ldots,N_v$, a total of $d_{(i)}$ copies of the corresponding vertex. Then, randomly choose pairs of elements from this list, removing each pair once chosen. The result will be a set of edges for a candidate graph G. If this set contains multi-edges or loops, the corresponding graph is a multi-graph, and so the set is discarded and the procedure repeated. Under appropriate conditions on the degree sequence, it can be argued that this algorithm will generate graphs from $\mathscr{G}$ with equal probability. See Molloy and Reed [282].

Unfortunately, in discarding ineligible candidate graphs, this algorithm can be quite inefficient. Indeed, it can be especially inefficient when the specified degree sequence is highly skewed, as with some of the examples seen in Section 4.2.1.1, because we can expect fairly frequently to see vertices of high degree matched with one another more than once. In fact, the frequency of such matchings can prevent the algorithm from completing with any sort of regularity. A common solution to this problem is to monitor the pairs of vertices being selected and, if a candidate pair matches one already removed from the list, it is rejected and another candidate selected instead. Note that this modification will necessarily introduce bias into the sampling, and the graphs G thus generated will no longer correspond to a strictly uniform sampling. Milo et al. [279], however, report a number of examples where this bias seems to be sufficiently small for practical purposes.

Alternatively, we can instead sample so as to avoid repeating existing matches in the first place. Chen, Diaconis, Holmes, and Liu [81], for example, introduce a method that is guaranteed to avoid repeated matches, due to a sequential assign-

ment of edges to each vertex in turn. Moreover, their method assigns these edges in a particularly efficient manner, through the use of an appropriately chosen non-uniform sampling scheme. The overall algorithm appears to produce nearly uniform samples in practice. The applications in their paper, however, are relatively small in scale, and it remains to be seen the extent to which their algorithm can successfully scale to large, sparse graphs.[7]

In contrast to matching algorithms, switching algorithms (also called *rewiring algorithms*) begin with a graph that has the prescribed degree sequence, and proceed to modify the connectivity of that graph through a succession of simple changes applied to randomly chosen subsets of edges. The term 'switching' derives from the nature of the basic change. Specifically, at each step, a pair of edges in the current graph, say $e_1 = \{u_1, v_1\}$ and $e_2 = \{u_2, v_2\}$, are randomly selected and replaced by the new edges $\{u_1, v_2\}$ and $\{u_2, v_1\}$, assuming neither of the latter already exists. That is, at each step of the algorithm it is proposed that a pair of randomly selected edges be 'switched'; but if either of the new edges that would result already exist, the proposed switch is abandoned.

This type of algorithm falls within the realm of MCMC algorithms. In practice, it is typical to let the algorithm run for some time before beginning to collect sample graphs G, so as to let the limiting behavior of the underlying Markov chain 'kick in.' But, as is often the case with MCMC methods, there is currently no well-developed theory to indicate just how long of a preliminary period is necessary. Milo et al. [279], however, cite empirical evidence to suggest a factor of $100N_e$ can be more than sufficient.

In order to ensure that the algorithm asymptotically yields strictly uniform sampling from $\mathscr{G}$, it is necessary that certain formal conditions be satisfied.[8] Rao, Jana, and Bandyopadhyay [322] describe such conditions for the basic switching algorithm described above and show that, in addition to the switching of pairs of edges, it is also necessary to allow for the possibility of a small number of three-edge exchanges.

MCMC methods, extending the basic switching concept described above, are also popular for generating generalized random graphs uniformly from other types of collections $\mathscr{G}$, in which additional characteristics beyond the degree sequence are constrained. This is largely due no doubt to the generality of the MCMC formalism, and the relative ease with which it may be implemented. Snijders [359], Roberts [326], and more recently, McDonald, Smith, and Forster [273], for example, offer MCMC algorithms for a number of different cases, such as where the graphs G

[7] The method of Chen, Diaconis, Holmes, and Liu [81] is actually developed for uniformly sampling $r \times c$ matrices $\mathbf{M}$ of non-negative integers with fixed marginal totals, such as can be needed in the analysis of contingency tables. However, it is immediately applicable to the problem of uniformly sampling generalized random graphs of fixed degree sequence, since the collection $\mathscr{G}$ can be linked in one-to-one correspondence with the collection $\mathscr{A}$ of all $N_v \times N_v$ binary adjacency matrices A with row and column sums matching the specified degree sequence. This duality has been exploited by a number of authors in this area. See Chen, Diaconis, Holmes, and Liu [81] for citations.

[8] In particular, the corresponding Markov chain should be irreducible on $\mathscr{G}$ and ergodic, so that convergence to the limiting uniform stationary distribution is guaranteed by the theory.

are directed and constrained to have both a fixed pair of in- and out-degree sequences and a fixed number of mutual arcs. It should be noted, however, that development of the corresponding theory, verifying the assumptions underlying Markov chain convergence, currently appears to lag far behind the pace of algorithm development.

6.2.4 Statistical Application of Random Graph Models

There are numerous ways in which random graph models can be used in statistical practice. We illustrate the possibilities here by describing their usage for model-based estimation of and assessment of significance of network graph characteristics.

6.2.4.1 Model-Based Estimation in Network Graphs

Recall the discussion in Example 6.1, where we revisited the problem of estimating a characteristic $\eta(G)$ of a network graph G, based on a sampled graph G^*. There were two aspects to the framework we described, with one aspect specifying the sampling mechanism used in producing G^* from G and the other specifying that G be the result of a uniform draw from a collection $\mathscr{G}$. The following example illustrates the use of such a framework, in the context of the problem of estimating the size of a hidden network population, as introduced in Example 5.7 of Chapter 5.

Example 6.3 (Estimating the Size of a 'Hidden Population' (Continued)). We describe a model-based derivation of the estimator in (5.43), again using a method-of-moments approach, and also due to Frank and Snijders [154]. Recall that the population graph G was a digraph, with vertices corresponding to a population of hidden individuals, and arcs, to the pattern of references among individuals if asked to identify other hidden individuals than themselves. The goal was to estimate the size of this population (i.e., $N_v = |V|$).

Suppose now that we model G as being from a collection $\mathscr{G}$ of random graphs, where each graph is constructed from an initial vertex set V by independently adding arcs between distinct pairs of vertices with probability $p = p_{\mathscr{G}}$. This is a directed version of the random graph model $\mathscr{G}_{N_v,p}$ introduced earlier in this chapter. Then, given G, we again assume that a subgraph G^* is obtained through a one-wave snowball sample, with the initial sample of vertices V_0^* selected through Bernoulli sampling on V with probability $p = p_0$.

As before, let $N = |V_0^*|$ be the size of the initial sample, M_1 the number of arcs among individuals in V_0^*, and M_2 the number of arcs pointing from individuals in V_0^* to individuals in V_1^*, where V_1^* is the set of individuals discovered in the first wave of sampling (i.e., the set of individuals discovered after the initial sample V_0^*). Under the assumptions on the model and sampling, it can be shown that

$$\mathbb{E}(N) = N_v\, p_0 \tag{6.4}$$

$$\mathbb{E}(M_1) = N_v\,(N_v - 1)\, p_0^2\, p_{\mathscr{G}} \tag{6.5}$$

$$\mathbb{E}(M_2) = N_v\,(N_v - 1)\,p_0\,(1 - p_0)\,p_{\mathscr{G}} \quad . \tag{6.6}$$

Compare these expressions to those in equations (5.40) through (5.42).

After setting the right-hand sides of each of the above equations equal to the observed values n, m_1, and m_2 of the arguments of the left-hand sides, some algebra yields the method-of-moments solutions

$$\hat{p}_0 = m_1/(m_1 + m_2) \tag{6.7}$$

$$\hat{p}_{\mathscr{G}} = m_1(m_1 + m_2)/n[(n-1)m_1 + nm_2] \tag{6.8}$$

$$\hat{N}_v = n(m_1 + m_2)/m_1 \tag{6.9}$$

for the parameters $p_0, p_{\mathscr{G}}$, and N_v. Note that these are the same estimates of p_0 and N_v as those in Example 5.7, derived from (5.40) through (5.42).

Frank and Snijders [154] also provide another estimator using this model, which differs from that just derived, based on maximization of a conditional likelihood. □

Such usage of classical random graph models is traditionally not uncommon in the social networks literature. On a related note, we point out that there are other ways to incorporate models into the inference of population graph parameters η. For example, instead of specifying a model for the graph G itself, we might instead specify a model directly for η. We have in fact already touched briefly on one such use of models in Section 4.2.1.1, in discussing methods to characterize the shape of a degree distribution. There, for certain broad degree distributions we saw that we might entertain simple power-law models like that in (4.1), or in particular, the linearized version in (4.2). Recall, however, that it was found that the fitting of such models to estimate power-law exponents can potentially be a rather delicate affair. Additionally, such approaches do not themselves inherently address effects of network sampling bias, like those mentioned in Example 5.9.

6.2.4.2 Assessing Significance in Network Graphs

Recall the discussion in Example 6.2, where we proposed to assess the significance of the value $\eta(G^{obs})$ of a network characteristic η for an observed graph G^{obs}, by constructing a reference distribution $\mathbb{P}_{\eta,\mathscr{G}}$ and examining how likely $\eta(G^{obs})$ is under this distribution. As we remarked there, an important practical issue is determining which $\mathscr{G}$ to use.

Suppose, for example, that the value $\eta(G^{obs})$ of interest is the number of distinct triangles in G^{obs}. If the graphs in $\mathscr{G}$ are not restricted to have the same number of edges N_e^{obs} as G^{obs}, then an apparent significance in the number of triangles in G^{obs} could result simply from comparing against graphs G with too few or too many edges to even possibly generate triangles at a similar order of magnitude. While this particular flaw can be easily remedied in this case, the basic point obviously stands. In practice, it is common to control for the degree sequence observed in

G^{obs}. However, at times there may arise other factors for which we might wish to control. For example, if we knew that certain groups were present within the vertex set V^{obs}, we might wish to maintain the number of edges between and within each group. We illustrate some of these various issues through the following example.

Example 6.4 (Assessing Clustering in the Karate Network). We saw in Section 4.3.1 that the karate club network has a clustering coefficient $\mathsf{cl}_T(G) = 0.2557$. With no other comparable sources of information, such as networks of similar clubs or dynamic snapshots of the same club, it is difficult to say *a priori* whether or not this value is in any sense unusual or unexpected. Therefore, we instead use random graphs to establish two abstract frames of reference, in the manner described above. For the first, we define $\mathscr{G}$ to be the collection of random graphs of the same order $N_v = 34$ and size $N_e = 78$ as the karate club network, while for the second, we add the further restriction that the elements of $\mathscr{G}$ have the same degree distribution as in the original.

For $\mathscr{G}$ defined with respect to a fixed number of edges, there are $\binom{\binom{34}{2}}{78} \approx 8.4 \times 10^{96}$ elements G. When $\mathscr{G}$ is instead defined through specifying the degree distribution, the total number of elements is much less, but still quite large. It is in fact a difficult problem in and of itself to evaluate this number, and essentially infeasible for graphs of order any larger than about $N_v = 10$. Discussion on this point may be found in Snijders [359] or Chen, Diaconis, Holmes, and Liu [81], for example. Therefore, we instead approximate the distribution in (6.2) by simulating[9] the uniform sampling of $10,000$ random graphs G from $\mathscr{G}$, for each of these two choices of $\mathscr{G}$, and calculating $\eta(G) = \mathsf{cl}_T(G)$ for each graph sampled.

Histograms showing the resulting distributions may be found in Figure 6.1. We see that under both random graph models it is quite unlikely to see a clustering coefficient as high as the observed value of 0.2557. In the first case, when the number of edges was fixed, only 3 of our $10,000$ samples resulted in a graph with higher clustering coefficient, whereas in the second case, when the degree distribution was fixed, all of the samples resulted in lower clustering coefficient. We therefore have strong evidence to reject the hypothesis that the karate club network can be viewed as a uniform sample under either random graph model. As a result, we conclude that the network graph shows markedly greater transitivity than random graphs of comparable magnitude (i.e., with respect to order and size) or connectivity (i.e., with respect to degree distribution).

We note that it is also of some interest to compare the two histograms in Figure 6.1. The distribution in the first case is noticeably more broad than that in the second, and shifted slightly more to the left. This indicates that further conditioning on degree distribution, rather than just the number of edges, has the effect of restricting to graphs for which the clustering coefficient is slightly higher and also more concentrated. In addition, note the suggestion of bimodal structure in the bottom histogram. Visual examination of sampled graphs from each of the two modes

[9] Simulations were conducted using Monte Carlo algorithms, implemented in the R package `statnet`. See *http://csde.washington.edu/statnet*.

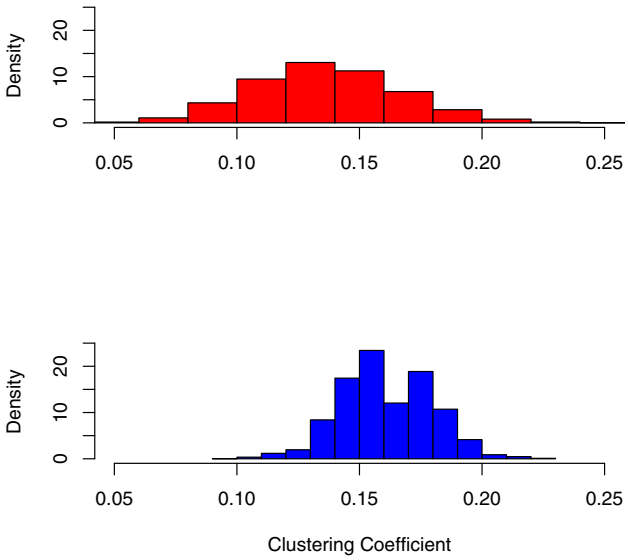

Fig. 6.1 Histograms of clustering coefficients $\mathsf{cl}_T(G)$ for random graphs of order $N_v = 34$, generated uniformly from those with the same number of edges N_e (top, in red) and the same degree distribution (bottom, in blue) as in the karate club network.

showed that, roughly speaking, those in the right-hand mode tended to often be characterized by two large and somewhat distinct clusters, as in the original network (see Figure 1.2), while those in the left-hand mode appeared to have more diffuse clusters or even just one large cluster. Finally, also worth mentioning is the observation that, due to the restriction placed on the network through conditioning on degree, coupled with the invariance of cl_T under isomorphism, the effective size of the sample space becomes quite small. In the $10,000$ trials run, there were only 25 different values of cl_T; in fact, over 99% of the mass of the distribution rested on just 17 distinct values. □

6.2.4.3 Detecting Network Motifs

A related use of random graph models is in the detection of *network motifs*, defined by Alon and colleagues [218, 278] to be small subgraphs occurring far more frequently in a given network than in comparable random graphs. The motivation for this interest in subgraphs is the idea that many large, complex networks may perhaps be constructed – at least in part – of smaller, comparatively simple 'building

blocks.' Network motif detection seeks to identify possible subgraph configurations of this nature.

In Milo et al. [278], each of the elements N_i of the vector $(N_1, \ldots, N_{L_k})$ are calculated, where N_i is the number of occurrences of the i-th of L_k possible types of k-vertex subgraphs. Similarly, in Kashtan, Itzkovitz, Milo, and Alon [218] the relative proportions

$$F_i = \frac{N_i}{\sum_{i'=1}^{L_k} N_{i'}} \tag{6.10}$$

are used. Then, analogous to what was described in the previous section, each value of N_i (or F_i) is compared to an appropriate reference distribution $\mathbb{P}_{N_i,\mathscr{G}}$ (or $\mathbb{P}_{F_i,\mathscr{G}}$) resulting from calculation of the same subgraph count (or relative count) statistic for all G in some collection $\mathscr{G}$. Subgraphs for whom the value N_i (or F_i) is found to be extreme are declared to be network motifs.

Example 6.5 (Detecting Motifs in the AIDS Blog Network). Consider the AIDS blog network in Figure 1.4 of Chapter 1. This directed graph represents $N_e = 183$ web-link-based citations (excluding loops and multi-edges) among $N_v = 146$ bloggers in a group focused on topics relating to AIDS. Although the network is not large, a simple glance at the figure is sufficient to see that it would be non-trivial to detect small, recurrent subgraph patterns by eye alone. The network motif methodology just described allows for an automated and principled approach to this task.

A total of $10,000$ random digraphs were generated using a switching algorithm, with in- and out-degree sequences, and also mutual edges, fixed to equal those of the original AIDS blog network.[10] Evidence for motifs of size $k = 3$ and 4 was examined, based on the appropriate reference distributions. Subgraphs for which the probability, under this random graph model, of seeing N_i as large or larger (i.e., the p-value) was 0.01 or smaller were declared to be motifs. One three-vertex subgraph and four four-vertex subgraphs were discovered using this criterion. These are displayed in Figures 6.2 and 6.3, respectively.

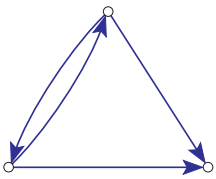

Fig. 6.2 Three-vertex motif discovered for the AIDS blog network.

[10] Calculations were done using the `mfinder1.2` software. See *http://www.weizmann.ac.il/mcb/UriAlon/groupNetworkMotifSW.html* .

There were a total of 2,442 three-vertex subgraphs found in our network graph. The three-vertex motif that we discovered occurred 15 times in this network, but only somewhat less than 7 times on average among the random graphs sampled (mean 6.7, with standard deviation 2.2). It represents two bloggers pointing to each other, with each in turn also pointing to a third blogger. That is, they share a mutual edge, and in addition both point to a common 'authority.' A total of 27,887 four-vertex subgraphs are contained in the network graph, and the four-vertex motifs were found to occur 704, 37, 13, and 5 times, respectively, based on the same criteria as described above for the three-vertex case. The first four-vertex motif represents one blogger pointing to another, who in turn points to two others. The other three four-vertex motifs are a bit more complicated in nature, but note that each of them contains within it one or more versions of the three-vertex motif and also the first four-vertex motif. So there is an argument to be made that the three-vertex motif and the first four-vertex motif are the most likely candidates to serve as elementary 'building block' structures.

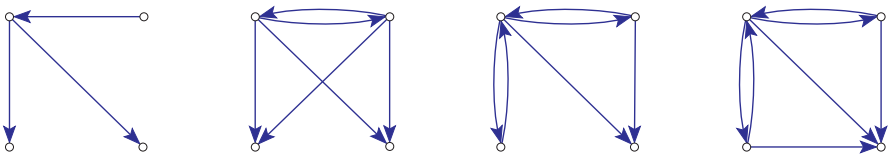

Fig. 6.3 Four-vertex motifs discovered for the AIDS blog network.

On a cautionary note, however, we point out that with the network graph as small as it is here, and the blog citations as inter-woven as they are, most of the individual copies of any of these motifs are overlapping with other versions of themselves. In larger networks, it is common to also require that a subgraph occur at some minimum number of disjoint locations in order to be declared a motif. □

Network motif detection is based on the basic principles of assessing significance we already discussed earlier. The new wrinkle here is primarily in the decision to focus on arbitrary subgraphs and their frequency of occurrence,[11] coupled with the determination to do this in large networks (e.g., $N_e > 100,000$) or for motifs defined by relatively large subgraphs (e.g., $k \geq 5$). With these larger problems come substantial computational challenges. For a given choice of k, we are nominally required to count the number N_i of each possible k-vertex subgraph, in both the observed network G^{obs} and each (or at least a large random subset) of the networks G in the collection $\mathscr{G}$. But the number L_k of possible motifs grows quite quickly in k. For example, in a directed graph G, there are $L_3 = 13$ distinct types of connected,

[11] We note that a precursor in the social network literature is the use of the so-called *triad census* in testing for group structure. See Wasserman and Faust [393, Sec. 14.4].

three-vertex subgraphs, $L_4 = 199$ such four-vertex subgraphs, and a rather daunting $L_5 = 9364$ such five-vertex subgraphs.

Graph sampling techniques, like those encountered in Chapter 5, are appealing for use in overcoming this computational challenge. Specifically, if k-vertex subgraphs H are sampled in some fashion, then an unbiased estimate of the total number N_i of a given subgraph type is just

$$\hat{N}_i = \sum_{H \text{ of type } i} \pi_H^{-1} \quad , \tag{6.11}$$

where π_H is the inclusion probability for H. Natural (although biased) estimates $\hat{F}_i$ of the corresponding relative frequencies F_i are obtained through direct substitution of (6.11) in (6.10).

Kashtan, Itzkovitz, Milo, and Alon [218] propose the following sampling method. Given a network graph G, we first randomly select a single edge. Then, using the idea of link-tracing introduced in Section 5.3.3, we follow a new edge to a neighboring vertex, where that edge is selected randomly from among those in G incident to the two vertices defining the first edge. We then continue in this fashion, where at the m-th stage we follow an edge randomly selected from among those incident to the collection of vertices at hand at the $(m-1)$-st stage, excluding those edges already encountered. When a total of k vertices have been sampled in this manner, we obtain the final sampled subgraph H by joining to these vertices and the edges encountered any remaining unencountered edges among these k vertices. That is, H is defined to be the induced subgraph on the set of discovered vertices.

Note that, using this sampling strategy, in order to sample a given k-vertex subgraph H, an ordered set of $k-1$ appropriate edges must be sampled. The probability of sampling H is therefore the sum of the probabilities of all such possible ordered edge sets. This probability is given by

$$p_H = \sum_{\sigma \in S} \prod_{e_j \in \sigma} \mathbb{P}\left(e_j \,|\, (e_1, \ldots, e_{j-1})\right) \quad , \tag{6.12}$$

where S is the set of all possible sets of $k-1$ ordered edges, σ is an element of that set, and e_j is the j-th edge in a specific element σ. The calculation of such probabilities is illustrated in Kashtan, Itzkovitz, Milo, and Alon [218, Fig. 2].

The probability that H is included in a collection of n subgraphs sampled in this fashion is just $\pi_H = 1 - (1 - p_H)^n$, following reasoning similar to that in Example 5.3. Kashtan et al. use the approximation $\pi_H \approx np_H$, which holds when a subgraph H is unlikely to be observed in the sample more than once at most. They also present an algorithm for the overall procedure, with an $O(k^2)$ complexity for sampling each subgraph H and an $O(C^{k-1}k^{k+1})$ complexity for calculating p_H, where C is a small constant correlated with the average degree $\bar{d}$ of the graph G being sampled. Numerical results show that this approach can produce substantial savings in computation time, while maintaining good levels of accuracy, in graphs G with broad degree distributions. However, with concentrated degree distributions, it appears the method does not provide similar savings.

6.3 Small-World Models

Arguably one of the more important innovations in modern network graph modeling is a movement from traditional random graph models, like those in Section 6.2, to models explicitly designed to mimic certain observed 'real-world' properties, often through the incorporation of a simple mechanism(s). Work of this type received a good deal of its impetus from the seminal paper of Watts and Strogatz [396] and the 'small-world' network model introduced therein.[12]

From an applications perspective, small-world network graph models are of interest across a wide range of disciplines. The interest in these models is largely due to their particular relevance to the topic of communication – where 'communication' should be interpreted in the broadest sense. The fact that a network is a small-world network essentially means that its structure is such that a system of neighbor-to-neighbor exchanges is sufficient to 'transmit' information quickly across the network, on average. Hearkening back to the context of the original experiment of Milgram [277], it is evident that small worlds are relevant to the spread of news, gossip, rumors, and the like. But they also have been found to be of importance in the spread of diseases, both natural – as in the case of human infectious diseases – and virtual – as in the Internet. See the book by Watts [395], for example, for a broad-ranging non-technical overview and commentary on small-world networks and their relevance. We will consider such topics in slightly more detail in Chapter 8, when we look at models for processes on network graph. Here, in this section, we concentrate simply on presenting definitions and basic properties of small-world network models.

6.3.1 The Watts-Strogatz Model

Watts and Strogatz [396] were intrigued by the fact that many networks in the real world display high levels of clustering, but small distances between most nodes. To better appreciate that this should seem a surprising combination, recall the case of the classical random graph model, for which it was observed in Section 6.2.1 that while the diameter varied as $O(\log N_v)$, indicating 'small-world' behavior, the clustering coefficient cl_T varied as N_v^{-1}, which suggests very little clustering.

In order to create a network graph with both of these properties, Watts and Strogatz suggested instead beginning with a graph with lattice structure, and then randomly 'rewiring' a small percentage of the edges. More specifically, in this model we begin with a set of N_v vertices, arranged in a periodic fashion, and join each vertex to r of its neighbors to each side. Then, for each edge, independently and with probability p, one end of that edge will be moved to be incident to another vertex,

[12] Bollobás and Riordan [45, Sec. 1.4] point out that mathematical treatments of analogous models of small-world phenomena can be found somewhat earlier in the literature on classical random graphs.

where that new vertex is chosen uniformly, but with attention to avoid the construction of loops and multi-edges. An example of a small-world network graph of this sort is shown in Figure 6.4.

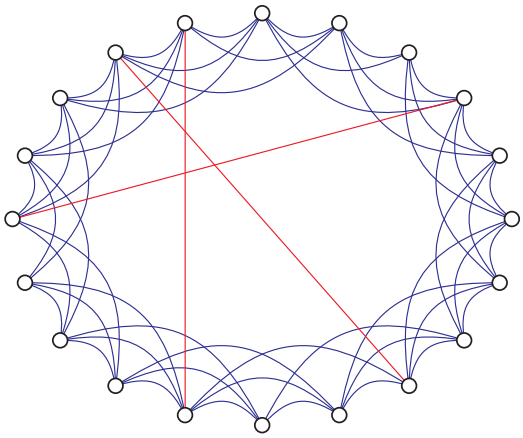

Fig. 6.4 Example of a Watts-Strogatz 'small-world' network graph. Blue edges pertain to the original underlying lattice; red edges are rewired.

For the lattice alone, it can be shown that the clustering coefficient $\mathsf{cl}_T(G) = (3r-3)/(4r-2)$, or roughly $3/4$ for r large, which indicates a high level of clustering. On the other hand, the diameter clearly varies like $N_v/(2r)$, and the average distance $\bar{l}$ behaves similarly as $N_v/(4r)$, for fixed r as N_v grows. So the distance between vertices on the lattice can be made to stay arbitrarily large, in defiance of the small-world property. But the addition of a few randomly rewired edges has the effect of producing 'short-cuts' in the graph. In numerical simulations, due to these short-cuts, Watts and Strogatz observed $\bar{l}$ to drop down to a magnitude of $O(\log N_v)$, even while the graph maintained a clustering coefficient close to $3/4$. This effect may be achieved even with very small p.

To illustrate, consider Figure 6.5, in which we show results from simulating a particular Watts-Strogatz small-world network model, with $N_v = 1,000$ and constant lattice degree of $d = 12$. Re-wiring is then done with probability p, as p varies from 0 to 1. Both the clustering coefficient[13] and the average distance $\bar{l}$ have been normalized by their largest values, which occur at a re-wiring probability of $p = 0$. It is quite evident from the figure that, over a non-trivial range of p – from roughly 10^{-3} to 10^{-1} in this particular setting – the network exhibits small average distance while maintaining a high level of clustering.

[13] Here we use cl, as in the original Watts-Strogatz analysis, rather than cl_T.

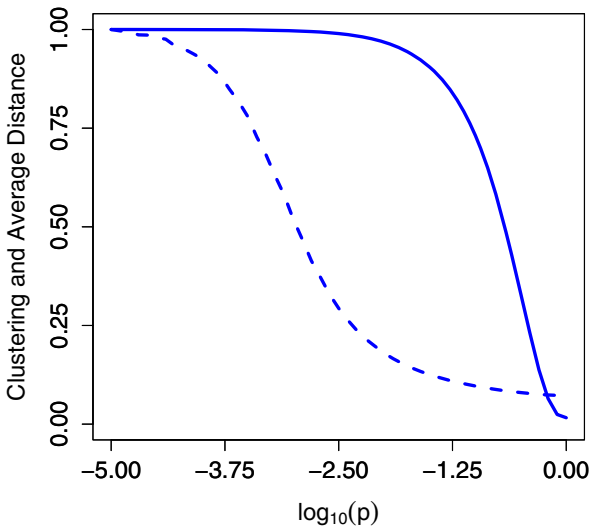

Fig. 6.5 Plot of the clustering coefficient $\mathsf{cl}(G)$ (solid) and average geodesic distance $\bar{l}$ (dashed), as a function of the rewiring probability p for a Watts-Strogatz small-world model. Results are averages based on 1,000 simulation trials.

Since the introduction of the model of Watts and Strogatz, a number of authors have focused upon better understanding its properties. However, the precise, formal characterization of these properties appears to still be an open problem. Barrat and Weigt [19] have argued that, for large N_v, the clustering coefficient cl_T should behave like

$$\frac{3r-3}{4r-2}(1-p)^3 \quad . \tag{6.13}$$

This expression differs from the value for the classical random graph only by the factor $(1-p)^3$, which will be relatively close to 1 for small p. Similarly, there has been progress on providing partial analytical characterizations of $\bar{l}$, in the form of partial series expansions and scaling properties. See Newman [296, Sec. VI] and Boccaletti et al. [41] for details and further references.

6.3.2 Other Small-World Network Models

Often it is slight variations of the original Watts-Strogatz model that are actually studied in the literature, being essentially equivalent and more amenable to analytical calculations. Perhaps most popular is the model where no edges are rewired, but

rather a small number of new edges are added to randomly selected pairs of vertices, as proposed by Monasson [284] and by Newman and Watts [298]. Note that in terms of simulation, any of these models are straightforward to generate. For example, the model in Newman and Watts [298] simply requires the generation of (i) a lattice, which may be done easily in linear time, and (ii) a classical random graph on top of that. Using the 'skipping' algorithm described in Section 6.2.3 for the second step, we obtain an overall $O(N_v + N_e)$ algorithm. Similarly, Batagelj and Brandes [23] provide an $O(N_v + N_e)$ algorithm for a version of the original small-world model in which edges are fully rewired (i.e., both vertices change).

Other mechanisms have also been proposed that successfully generate network graphs with the small-world property. For example, Kasturirangan [219] begins with a one-dimensional periodic lattice, but then, instead of rewiring or adding edges among vertices in that lattice, a small number of vertices are added to the center and each is connected randomly to a large number of vertices throughout the original lattice. Alternatively, Kleinberg [229, 231], studying small-world network graphs from an algorithmic perspective, proposes a model that begins with a two-dimensional lattice (i.e., a standard grid) and then adds short-cuts between vertices with a probability varying inversely proportional to the distance between them. That is, with $p \propto (\mathsf{dist})^{-r}$, for some $r > 0$. Kleinberg shows that, whereas for the original Watts-Strogatz model no algorithm exists which can find shortest paths between vertices using only local information, for his model, when $r = 2$ (and, interestingly, only when $r = 2$), there exists a simple algorithm for doing so. The implication here is that, in order to mimic the types of small worlds that we navigate successfully on a daily basis, it is not sufficient simply to add random short-cuts. Rather, there must be some bias in the way they are laid, encouraging them to be – at an appropriate level – more local than not.

6.4 Network Growth Models

Many networks grow or otherwise evolve in time. The World Wide Web and scientific citation networks are two obvious examples. Similarly, many biological networks may be viewed as evolving as well, over appropriately defined time scales. Much energy has been invested in the development of models that mimic network growth.[14] In this arena, typically a simple mechanism(s) is specified for how the network changes at any given point in time, based on concepts like vertex preference, fitness, copying, age, and the like. Usually interest then centers on what properties emerge for the network in the limit of a large number of consecutive time periods. If certain properties are found to match those observed in real-world networks, this is often taken as suggestive that the specified mechanism is perhaps a reasonable approximation to a similar, real-world mechanism. Although generally the reality of the forces behind the emergence of real-world networks is decidedly more subtle

[14] What we call 'network growth' models here are commonly referred to as 'evolutionary network' models in the physics literature.

and involved than the comparatively simple mechanisms proposed in these models, work on mechanisms like these nevertheless has proven useful in thinking more deeply about real-world network growth and design.

In this section, we discuss models based on two fundamental and popular classes of growth mechanisms – preferential attachment and copying. We also briefly discuss the problem of fitting such models to network data.

6.4.1 Preferential Attachment Models

Preferential attachment mechanisms are designed to embody the principle that 'the rich get richer.' A driving motivation behind the introduction of these mechanisms was a desire to reproduce the types of broad degree distributions observed in many large, real-world networks. Simon [353], intrigued by the appearance of power-law distributions in various settings (e.g., the frequency of occurrence of words, or the relative size of city populations), proposed a class of models in 1955 that produced such broad, skewed distributions. Price [112] then took this idea a decade later and applied it in creating a model for the manner in which networks of citations for documents in the literature grow. But it is the work of Barabási and Albert [18], nearly 35 years after Price, that launched the present-day fascination with models of this type, and so it is with their model that we begin our exposition here.[15]

Barabási and Albert were motivated by the growth of the World Wide Web, noting that often web pages to which many other pages point will tend to accumulate increasingly greater numbers of links as time goes on. For example, think of how people increasingly added websites like Google to their list of 'Favorites' in their web browsers, as those sites grew in popularity. The Barabási-Albert (BA) model is a network growth model for undirected graphs, formulated originally as follows. Start with an initial graph $G^{(0)}$ of $N_v^{(0)}$ vertices and $N_e^{(0)}$ edges. Then, at stage $t = 1, 2, \ldots$, the current graph $G^{(t-1)}$ is modified to create a new graph $G^{(t)}$ by adding a new vertex of degree $m \geq 1$, where the m new edges are attached to m different vertices in $G^{(t-1)}$, and the probability that the new vertex will be connected to a given vertex v is given by

$$\frac{d_v}{\sum_{v' \in V} d_{v'}} \quad . \tag{6.14}$$

That is, at each stage, m existing vertices are connected to a new vertex in a manner preferential to those with higher degrees. After t iterations, the resulting graph $G^{(t)}$ will have $N_v^{(t)} = N_v^{(0)} + t$ vertices and $N_e^{(t)} = N_e^{(0)} + tm$ edges. And, because of the tendency towards preferential attachment, intuitively we would expect that a number of vertices of comparatively high degree should gradually emerge as t increases.

Note that in this formulation there is some latitude left for specifying precisely how to select a set of m vertices in proportion to their degree, since there is in

[15] For an extensive history of power-law models, including their use in network modeling, see Mitzenmacher [280].

fact more than one way to do so.[16] Bollobás and colleagues [44, 46], citing this ambiguity, have proposed a particular version of the BA preferential attachment model in which the nature of the sampling is made precise, called the *linearized-chord diagram* (LCD) model, which is defined as follows. For the case $m = 1$, we begin with a graph $G^{(1)}$ consisting of a single vertex with a loop. Then, for each $t = 2, 3, \ldots$, the graph $G^{(t)}$ is created from $G^{(t-1)}$ by adding the vertex v_t to $G^{(t-1)}$ with an edge to a vertex v_s, for $1 \leq s \leq t$ chosen randomly with respect to the distribution

$$\mathbb{P}(s = j) = \begin{cases} d_{G^{(t-1)}}(v_j)/(2t-1), & \text{if } 1 \leq j \leq t-1 \ , \\ 1/(2t-1), & \text{if } j = t \ , \end{cases} \tag{6.15}$$

where $d_{G^{(t-1)}}(v_j)$ denotes the degree of the vertex v_j at time $t-1$. For the case $m > 1$, the above process is simply run for m steps at a time, after which the m vertices just created are contracted into one vertex, with the m edges retained. Note that this formulation of the preferential attachment model allows for loops and multi-edges. However, these should occur relatively rarely, and the precision gained by this formulation is necessary for rigorously deriving mathematical results regarding model properties.

The most celebrated property of such preferential attachment models is that, in the limit as t tends to infinity, the graphs $G^{(t)}$ have degree distributions that tend to a power-law form $d^{-\alpha}$, with $\alpha = 3$. More precisely, formalizing a body of earlier results, such as Barabási and Albert [18], Dorogovstev, Mendes, and Samukhin [124], and Krapivsky, Redner, and Leyvraz [240], it was shown in Bollobás, Riordan, Spencer, and Tusnady [46] that with high probability

$$(1-\varepsilon)\, f_{d,m} \leq f_d(G^{(t)}) \leq (1+\varepsilon)\, f_{d,m} \ , \tag{6.16}$$

for any $\varepsilon > 0$, and every d in the range $0 \leq d \leq \left(N_v^{(t)}\right)^{1/5}$, where

$$f_{d,m} = \frac{2m(m+1)}{(d+2)(d+1)d} \ , \tag{6.17}$$

which behaves like d^{-3} for d large relative to m. The expression in (6.16) may be compared to that in (6.3), describing the Poisson limiting behavior for classical random graphs.

A number of other properties of interest have been established for these preferential attachment models as well. Note that in the original formulation of the BA model, the graphs $G^{(t)}$ are connected, by definition, assuming that the initial graph $G^{(0)}$ was connected. But in the LCD model, these graphs are no longer necessarily connected. However, Bollobás and Riordan [44] show that they are connected with high probability. They also show that these random graphs will exhibit small-world behavior. In particular, with high probability, in the case $m = 1$, the diam-

[16] See Bollobás and Riordan [45, Sec. 1.6] for a discussion of some of the issues that can arise due to this latitude.

eter of $G^{(t)}$ behaves like $O(\log N_v^{(t)})$; in the case $m > 1$, it actually behaves like $O(\log N_v^{(t)} / \log\log N_v^{(t)})$, which is a bit smaller still. On the other hand, with respect to clustering, these models are known to be somewhat unsatisfactory, in that the resulting graphs display decidedly less clustering than often observed in real-world networks of similar magnitude. For instance, Bollobás and Riordan [45] show that the expected value $\mathbb{E}[\mathsf{cl}_T(G^{(t)})]$ behaves like

$$\frac{m-1}{8}\,\frac{(\log N_v^{(t)})^2}{N_v^{(t)}} \quad , \tag{6.18}$$

which is only a little better than the N_v^{-1} behavior in the case of classical random graph models.

These preferential attachment models may be simulated in linear time. Batagelj and Brandes [23] describe an efficient algorithm for simulating the LCD model. Also see Newman [296, Sec. VII.B].

There have been many extensions and variations on the basic preferential attachment model that have been proposed in the literature – far too many to even summarize adequately here – for modeling both undirected graphs (as in the original BA model) and directed graphs. However, overall the common concern in offering these extensions has primarily been (i) whether or not a power-law limiting distribution is achieved, and (ii) if so, the manner in which the model mechanism(s) and parameters affect the power-law exponent. Some models modify the functional form of the preferential attachment probability in (6.14), and by doing so produce a richer collection of possible limiting behaviors for the degree distribution of the resulting random graphs than the original BA model. For example, even simply including an offset parameter $-m < d^* < \infty$ in the numerator and denominator, by changing d_v to $d_v + d^*$ throughout, yields models with limiting degree distributions that behave like power laws with exponent $\alpha = 3 + d^*/m$. That is, we can produce exponents anywhere in the range $(2, \infty)$. Or, replacing the linear role of the degrees d_v in (6.14) by powers of the form d_v^γ, it is possible to produce limiting degree distributions with a number of different functional forms, as γ varies, deviating to various extents from the strict power-law behavior arising in the case $\gamma = 1$. Models including some measure of the 'fitness' or inherent quality of vertices have also been proposed by many authors, as have models that supplement preferential attachment with some additional mechanism(s), often with an eye towards inducing a greater amount of clustering in the network. Finally, models have also been proposed that allow the number of new edges m to vary randomly, as well as models that allow for both addition and removal of edges.

See Albert and Barabási [6] for an extensive review of much of this literature, and their Table III for a concise summary. Also see Newman [296, Sec. VII.C] and Boccaletti et al. [41, Sec. 2.3.5], which offer summaries a bit more recent, albeit less extensive. Additional resources are also described in Section 6.7.

6.4.2 Copying Models

Another mechanism of fundamental interest, which is distinct from preferential attachment but can nevertheless also produce power-law degree distributions, is that of copying. Like preferential attachment, *copying mechanisms* were also originally proposed in the context of the World Wide Web – in this case by Kleinberg et al. [232]. However, they arguably have generated the most interest in the context of biochemical networks where, unlike preferential attachment, copying is felt to be a tenable mechanism from the perspective of basic biology. For example, it is commonly held that gene duplication is at the heart of nature's observed tendency to 're-use' biological information in evolving the genomes of living organisms (e.g., Ohno [305]).

Chung, Lu, Dewey, and Galas [89] offer a network growth model based on the following simple, straightforward notion of copying (or 'duplication,' in their terminology). Beginning with an initial graph $G^{(0)}$, graphs $G^{(t)}$ are constructed from their immediate predecessors, $G^{(t-1)}$, by the addition of a new vertex, say v, that is connected to some randomly chosen subset of neighbors of a randomly chosen existing vertex, say u. More precisely, a vertex u is chosen from $G^{(t-1)}$ uniformly at random, and then the new vertex v is joined with each of the neighbors of u in $G^{(t-1)}$ independently with probability p.

Chung et al. show that, with high probability, the degree distribution $\{f_d(G^{(t)})\}$ will tend to a power-law form $d^{-\alpha}$, with exponent α satisfying the equation

$$p(\alpha-1) = 1 - p^{\alpha-1} \quad . \tag{6.19}$$

This equation will have two solutions α for any given p, but only one will be stable. For $p > 0.5671\ldots$, the stable solution is simply $\alpha = 1$; for $p = 1/2$, it is $\alpha = 2$. Figure 6.6 shows a plot of the solutions, as a function of p.

Note that when $p = 1$, each new vertex v is connected to $G^{(t-1)}$ by fully duplicating the edges of the randomly selected vertex u. Citing previous numerical work of theirs showing that power-law behavior will not result when $p = 1$, Chung et al. prove that to achieve power-law behavior with full duplication it is sufficient to allow partial duplication to occur some fraction $q \in (0,1)$ of the times that a new vertex is added.

There have been other copying models suggested as well. For example, prior to Chung, Lu, Dewey, and Galas [89], growth models were proposed by Kleinberg et al. [232] and by Kumar et al. [241] in which, when creating $G^{(t)}$ from $G^{(t-1)}$, a new vertex v receives $m \geq 1$ new edges to vertices in $G^{(t-1)}$ by a combination of either (i) copying edges from a randomly chosen vertex u or (ii) simply establishing new edges. More precisely, for each new vertex, a 'prototype' vertex u is chosen from $G^{(t-1)}$, as in the approach of Chung, Lu, Dewey, and Galas [89]. Then, for $i = 1,\ldots,m$, the i-th edge is chosen with probability β to be from v to some vertex selected uniformly at random from $G^{(t-1)}$, while with probability $1-\beta$, it is chosen instead to be to the i-th neighbor of u.

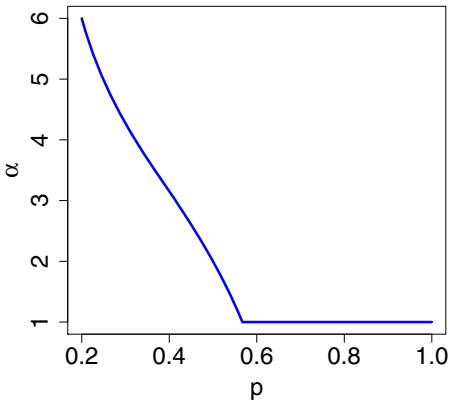

Fig. 6.6 Power-law exponent α, as a function of p, for the copying model of Chung, Lu, Dewey, and Galas [89], as given by the expression in equation (6.19).

The original conceptual image behind this model is that of an individual setting up a new web page. On the one hand, the web page is likely being set up in the context of some established topic(s), and thus will probably include a number of established links in that topic. On the other hand, an individual also brings their own perspective to the topic, and will be expected to generate some previously unseen links as well. Alternatively, as described in Kim, Krapivsky, Kahng, and Redner [226], where a very similar model is proposed for protein interaction networks, we can envision the structure of genes associated with the proteins as typically evolving through duplication, as above, but periodically arising instead through mutation.

For this model, the degree distribution $\{f_d(G^{(t)})\}$ also tends to a power law. Specifically, Kumar et al. [241] show that, for each $d > 0$, the sequence of values $f_d(G^{(t)})$ tends – as t tends to infinity – to

$$f_d = f_0 \prod_{j=1}^{d} \frac{1+\beta/(j(1-\beta))}{1+2/(j(1-\beta))} \quad . \tag{6.20}$$

The value f_d in (6.20) in turn can be shown to behave like $d^{-\alpha}$, for $\alpha = (2-\beta)/(1-\beta)$. Kumar et al. [241] also show that this model generates many more dense bipartite subgraphs than are found in comparable classical random graphs. This characteristic is encountered in real web graphs, and was one of the primary motivations behind their work.

On a final note, we mention that models have also been proposed that combine both copying and preferential attachment mechanisms. Cooper and Frieze [99], for example, introduce a particularly general network growth model of this sort and

establish a set of precise conditions for power-law behavior to occur. Their results, however, are somewhat complicated to state, and are omitted here.

6.4.3 Fitting Network Growth Models

By far the most common practical usage of network growth models (i.e., in conjunction with analyzing observed network graphs) has been predictive, in the sense of making informal comparisons between certain characteristics of an observed network and the graphs $G^{(t)}$ resulting from such models. Both theoretical calculations, like those described above, and numerical simulations are generally used in making comparisons of this sort. We note, however, that there has been little attempt to date to actually fit standard network growth models to network data. This is to be expected, on the one hand, given the comparative (and, generally, intentional) simplicity of such models. On the other hand, it can be argued that fitting even these simple models may be useful in some cases, in that estimates of parameters, like the duplication probability p in the copying model of Chung, Lu, Dewey, and Galas [89], can provide at least a rough characterization of basic rates and proportions of some fundamental interest.

Ideally, in order to fit a network growth model we would like to have a dynamic sequence of 'snap-shots' of the graph as it has changed over consecutive periods of time. In practice, however, as mentioned earlier in this book, such data still remain fairly elusive in most settings. Instead, we effectively find ourselves faced in most situations with a single snap-shot, which we may conveniently picture as the end-product of some sort of process of evolution. For this situation, Wiuf, Brameier, Hagberg, and Stumpf [407] have proposed a methodology for fitting network growth models that makes clever usage of their recursive nature to get around this lack of multiple snap-shots.

Suppose that the class of network growth models to be fit is defined so that a single vertex is added to $G^{(t-1)}$ in creating $G^{(t)}$ and, importantly, that the manner in which this vertex is added depends only on the vertices and edges present in $G^{(t-1)}$ and no earlier.[17] Such is the case, for example, in all of the models described above. Next, define $\delta(G^{(t)}, v)$ to be the graph obtained by deleting the vertex v and all edges incident to it from $G^{(t)}$. We will call v *removable* if $G^{(t)}$ can be created from $\delta(G^{(t)}, v)$ through copying. If $G^{(t)}$ contains no removable vertices, we will call it *irreducible*; otherwise, $G^{(t)}$ will be said to be *reducible.*

Wiuf et al. consider a class of models they call *duplication-attachment* (DA) models, parameterized by a vector θ, that includes a number of the models already described above as special cases. The key properties satisfied by these models are

(i) the initial graph $G^{(0)}$ is irreducible, and

(ii) $\mathbb{P}_\theta(G^{(t)}|G^{(t-1)}) > 0$ if and only if $G^{(t)}$ can be obtained by copying one vertex in $G^{(t-1)}$.

[17] That is, it is assumed that $\{G^{(t)}\}_{t=0}^{\infty}$ forms a Markov chain.

Assuming that $G^{(t)}$ represents the observed network graph, they define a likelihood function[18] for θ recursively as

$$\mathscr{L}\left(\theta; G^{(t)}\right) = \frac{1}{t} \sum_{v \in \mathscr{R}(G^{(t)})} \mathbb{P}_\theta\left(G^{(t)} \mid \delta(G^{(t)}, v)\right) \mathscr{L}\left(\theta; \delta(G^{(t)}, v)\right) \quad , \tag{6.21}$$

where $\mathscr{R}(G^{(t)})$ is the set of all removable nodes in $G^{(t)}$. A maximum likelihood estimate $\hat{\theta}$ is then defined as the maximizer of (6.21) in θ.

Unfortunately, explicit calculation of the likelihood function is non-trivial, even for network graphs of modest size, due to the recursion involved. For the extreme case of a complete graph, the complexity of this recursion varies like $O(t2^t)$; but in practice, even with sparser network graphs, Wiuf et al. observed the complexity to still scale quite poorly.

But Monte Carlo methods (i.e., so-called sequential importance sampling) may be used profitably to produce stochastic approximations to the likelihood. Wiuf et al. show how (6.21) may be re-expressed as an expectation in the form

$$\mathscr{L}\left(\theta; G^{(t)}\right) = \mathbb{E}_{\theta_0}\left[\prod_{s=0}^{t} S(\theta_0, \theta, G^{(s)}, v)\right] \quad , \tag{6.22}$$

for arbitrary θ_0, where $S(\cdot)$ is defined to be 1 at $s = 0$ and is a function of the probabilities $\mathbb{P}_{\theta'}(G^{(s)} | \delta(G^{(s)}, v))$ for $1 \leq s \leq t$ and $\theta' = \theta_0$ or θ. Sequences $G^{(1)}, \ldots, G^{(t)}$ are simulated according to the specified DA model, under θ_0, and $G^{(0)}$ is taken to be the irreducible graph underlying the observed graph. The product in (6.22) is evaluated for each simulated sequence. Averaging a sufficient number of these products yields an approximation to the expectation in (6.22), and hence to the likelihood function evaluated at θ.

Wiuf et al. illustrate the potential of this approach by calculating a univariate likelihood function for the probability p in an extended version of the copying model of Chung, Lu, Dewey, and Galas [89], for a network of interactions among 2,368 proteins in the worm organism *Caenorhabditis elegans.* However, as pointed out by these authors themselves, there are a number of issues that remain to be solved for the methodology to be scaled up effectively to more complicated contexts, such as those involving multivariate parameters, larger networks, and more realistic network growth models. It is also an open problem to establish an appropriate theory for confidence intervals and testing in this context.

[18] In principle, to specify a full likelihood function we should model the initial graph $G^{(0)}$. Wiuf et al. show that the set of irreducible graphs obtainable from repeatedly removing removable vertices forms an equivalence class, in the sense that all such graphs are isomorphic to each other, and so argue that the additional model structure is unnecessary. Alternatively, (6.21) may be viewed as defining a conditional likelihood.

6.5 Exponential Random Graph Models

The models discussed so far in this chapter serve a variety of useful purposes. Yet for the purpose of statistical model building, in the sense of proposing, fitting, and evaluating a model(s) in a manner informed intimately by observed data, they come up short. Indeed, as Robins and Morris [328] write, "A good [statistical network graph] model needs to be both estimable from data and a reasonable representation of that data, to be theoretically plausible about the type of effects that might have produced the network, and to be amenable to examining which competing effects might be the best explanation of the data." None of the models we have seen up until this point are really intended to meet such criteria.

In contrast, *exponential random graph models* (ERGMs)[19] are designed precisely with these criteria in mind and are formulated in a manner that facilitates the adaptation and extension of well-established statistical principles and methods for the construction, fitting, and comparison of models. As a result, they have the potential to come substantially closer to satisfying the criteria stated above. On the other hand, the highly complex patterns of dependency underlying the relational ties in typical observed network graphs mean that the necessary adaptations and extensions are far from trivial.

We discuss here the key aspects of of this approach to network graph modeling and point out some of its current limitations as well. An illustration of the methodology will be presented in the case study that follows this discussion, in Section 6.5.4.

6.5.1 Model Specification

An arbitrary (discrete) random vector $\mathbf{Z}$ is said to belong to an *exponential family* if its probability mass function may be expressed in the form

$$\mathbb{P}_\theta(\mathbf{Z} = \mathbf{z}) = \exp\left\{\theta^T \mathbf{g}(\mathbf{z}) - \psi(\theta)\right\} \quad , \tag{6.23}$$

where $\theta \in \mathbb{R}^p$ is a $p \times 1$ vector of parameters, $\mathbf{g}(\cdot)$ is a p-dimensional function of $\mathbf{z}$, and $\psi(\theta)$ is a normalization term, ensuring that $\mathbb{P}_\theta(\cdot)$ sums to one over its range. The class of discrete exponential families includes many familiar distributions, such as the binomial, geometric, and Poisson. In the case of continuous exponential families, where an analogous form of (6.23) holds for probability density functions, examples include the Gaussian and chi-square distributions. Exponential families all have have a variety of common useful algebraic and geometric properties, mak-

[19] These models are also referred to as p^* models, particularly in the social network literature, where they are seen as one of the later examples of a series of model classes introduced in succession over a roughly 20-year period covering the late 1970's, 1980's, and early 1990's. See the review of Wasserman and Pattison [394], for example. Our use of the term 'exponential random graph models' reflects current practice, which emphasizes the connection of these models with traditional exponential family models in classical statistics, as we shall see below.

ing this class of distributions mathematically convenient for purposes of inference and simulation.

Consider $G = (V, E)$ as a random graph. Let $Y_{ij} = Y_{ji}$ be a binary random variable indicating the presence or absence of an edge $e \in E$ between the two vertices i and j in V. The matrix $\mathbf{Y} = [Y_{ij}]$ is thus the (random) adjacency matrix for G. Denote by $\mathbf{y} = [y_{ij}]$ a particular realization of $\mathbf{Y}$. An exponential random graph model is a model specified in exponential family form for the joint distribution of the elements in $\mathbf{Y}$. More precisely, an ERGM takes the form

$$\mathbb{P}_\theta(\mathbf{Y} = \mathbf{y}) = \left(\frac{1}{\kappa}\right) \exp\left\{\sum_H \theta_H \, g_H(\mathbf{y})\right\} \quad , \tag{6.24}$$

where

(i) each H is a *configuration*, which is defined to be a set of possible edges among a subset of the vertices in G;
(ii) $g_H(\mathbf{y}) = \prod_{y_{ij} \in H} y_{ij}$, and is therefore either one if the configuration H occurs in $\mathbf{y}$, or zero, otherwise;
(iii) a non-zero value for θ_H means that the Y_{ij} are dependent for all pairs of vertices $\{i, j\}$ in H, conditional upon the rest of the graph; and
(iv) $\kappa = \kappa(\theta)$ is a normalization constant,

$$\kappa(\theta) = \sum_{\mathbf{y}} \exp\left\{\sum_H \theta_H \, g_H(\mathbf{y})\right\} \quad . \tag{6.25}$$

The summation in (6.24) is over all possible configurations H. Note that this model implies a certain (in)dependency structure among the elements in $\mathbf{Y}$. Generally speaking, such assumptions specify that the random variables $\{Y_{ij}\}_{(i,j)\in\mathscr{A}}$ are independent of $\{Y_{i'j'}\}_{(i',j')\in\mathscr{B}}$, conditional on the values of $\{Y_{i''j''}\}_{(i'',j'')\in\mathscr{C}}$, for given index sets $\mathscr{A}, \mathscr{B}$, and $\mathscr{C}$. Conversely, we can begin with a collection of (in)dependence relations among subsets of elements in $\mathbf{Y}$ and try to develop a model. However, it is important to realize that it is *not* the case that simply any collection of (in)dependence relations among the elements of $\mathbf{Y}$ yields a proper joint distribution on $\mathbf{Y}$. Rather, certain conditions must be satisfied, as formalized in the celebrated Hammersley-Clifford theorem (e.g., Besag [36]).

We postpone a somewhat more detailed discussion of these issues until Chapter 10, and instead content ourselves here with illustrating them through examples.

Example 6.6 (Bernoulli Random Graphs). Suppose we specify that, for any given pair of vertices, the presence or absence of an edge between that pair is independent of the status of possible edges between any other pairs of vertices. That is, for each pair $\{i, j\}$, we assume that Y_{ij} is independent of $Y_{i'j'}$, for any $\{i', j'\} \neq \{i, j\}$. This assumption implies that $\theta_H = 0$ for all configurations H involving three or more vertices. In this case, therefore, the only relevant functions g_H are those of the form $g_H(\mathbf{y}) = g_{ij}(\mathbf{y}) = y_{ij}$, and the ERGM in (6.24) thus reduces to

$$\mathbb{P}_\theta(\mathbf{Y}=\mathbf{y}) = \left(\frac{1}{\kappa}\right) \exp\left\{\sum_{i,j} \theta_{ij} y_{ij}\right\} \quad . \tag{6.26}$$

Note that (6.26) is just another way of writing that each edge $\{i,j\}$ is present in the graph independently with probability

$$p_{ij} = \exp(\theta_{ij})/[1+\exp(\theta_{ij})] \quad . \tag{6.27}$$

However, this entails a model with N_v^2 parameters, which is likely far too over-parameterized. In order to reduce the total number of parameters, it is common to impose an assumption of homogeneity across certain vertex pairs. For example, assuming homogeneity across all of G (i.e., $\theta_{ij} \equiv \theta$, for all $\{i,j\}$) yields

$$\mathbb{P}_\theta(\mathbf{Y}=\mathbf{y}) = \left(\frac{1}{\kappa}\right) \exp\{\theta L(\mathbf{y})\} \quad , \tag{6.28}$$

where $L(\mathbf{y}) = \sum_{i,j} y_{ij} = N_e$ is the number of edges in the graph. In this case, the Bernoulli random graph model of Section 6.2.1 is recovered, with $p = \exp(\theta)/[1+\exp(\theta)]$.

Alternatively, suppose that vertices are known *a priori* to fall within either of two sets, say S_1 and S_2. If we impose homogeneity within and between sets, we arrive at a model of the form

$$\mathbb{P}_\theta(\mathbf{Y}=\mathbf{y}) = \left(\frac{1}{\kappa}\right) \exp\{\theta_{11} L_{11}(\mathbf{y}) + \theta_{12} L_{12}(\mathbf{y}) + \theta_{22} L_{22}(\mathbf{y})\} \quad , \tag{6.29}$$

where $L_{11}(\mathbf{y})$ and $L_{22}(\mathbf{y})$ are the number of edges within sets S_1 and S_2, respectively, and $L_{12}(\mathbf{y})$ is the number of edges between S_1 and S_2. □

Unfortunately, assumptions of complete independence among possible edges are largely untenable in practice. In addition, we saw ample evidence earlier, in Section 6.2.1, to indicate that Bernoulli-like random graphs lack the ability to reproduce many of the most basic structural characteristics observed in most real-world networks. But certain simple conditional independence assumptions may be used profitably to create a much richer class of models.

Example 6.7 (Markov Random Graphs). Frank and Strauss [155] introduced the notion of *Markov dependence* for network graph models, which specifies that two possible edges are dependent whenever they share a vertex, conditional on all other possible edges. That is, the presence or absence of $\{i,j\}$ in the graph will depend upon that of $\{i,k\}$, for a given $k \neq j$, even given information on the status of all other possible edges in the network. A random graph G arising under Markov dependence conditions is called a *Markov graph*.

Under an assumption of homogeneity, using the Hammersley-Clifford theorem, Frank and Strauss show that G is a Markov graph if and only if $\mathbb{P}_\theta(\cdot)$ may be expressed as

$$\mathbb{P}_\theta(\mathbf{Y}=\mathbf{y}) = \left(\frac{1}{\kappa}\right) \exp\left\{\sum_{k=1}^{N_v-1} \theta_k S_k(\mathbf{y}) + \theta_\tau T(\mathbf{y})\right\} \quad , \tag{6.30}$$

where $S_1(\mathbf{y}) = N_e$ is the number of edges, $S_k(\mathbf{y})$ is the number of k-stars,[20] for $2 \leq k \leq N_v - 1$, and $T(\mathbf{y})$ is the number of triangles.

Note that the statistics S_k in (6.30), and also T, can be expected to often be correlated. For example, more edges in G clearly allows for the possibility of more k-stars of certain orders. Similarly, more k-stars, for a given k, means more k'-stars, for $k' < k$. In this sense, we may view the θ_k, for increasingly larger k, as successively higher-order effects in the model. And the inclusion of lower-order effects $\theta_{k'}$ means that θ_k is the effect due to k-stars, adjusted for the quantity of lower-order stars. Additionally, more k-stars means the potential for more triangles, and the value θ_τ is therefore the effect of triangles, adjusted for the levels of all k-stars.

Interestingly, we see from the nature of the statistics S_k and T that the transition from the independence assumptions underlying Bernoulli models to Markov dependence results in a model explicitly parameterized to account for some effects of transitivity, something lacking in the models introduced previously in this chapter. □

In using Markov graph models, it has traditionally been common practice to include star counts S_k no higher than $k = 2$, or at most $k = 3$, by setting $\theta_4 = \cdots = \theta_{N_v-1} = 0$. But experience has shown this practice to frequently produce models that fit quite poorly to real data. Investigation of this phenomena has found it to be intimately related to the issue of model degeneracy, which we define and discuss below in Section 6.5.3. Unfortunately, the alternative – including a sufficiently large number of higher order terms – is problematic as well, from the perspective of model fitting.

A solution to this dilemma, proposed by Snijders, Pattison, Robins, and Handcock [360], is to impose a parametric constraint of the form $\theta_k \propto (-1)^k \lambda^{2-k}$ upon the star parameters, for all $k \geq 2$, for some $\lambda \geq 1$. This tactic has the effect of combining all of the k-star statistics $S_k(\mathbf{y})$ in (6.30), for $k \geq 2$, into a single *alternating k-star statistic* of the form

$$AKS_\lambda(\mathbf{y}) = \sum_{k=2}^{N_v-1} (-1)^k \frac{S_k(\mathbf{y})}{\lambda^{k-2}} \quad , \tag{6.31}$$

and weighting that statistic by a single parameter θ_{AKS} that takes into account the star effects of all orders simultaneously.

One may think of the alternating signs in (6.31) as allowing the counts of k-stars of successively greater order to balance each other, rather than simply ballooning. Alternatively, it may be shown that the statistic $AKS_\lambda(\mathbf{y})$ is a linear function of the *geometrically weighted degree count,* which is defined as

[20] A k-star is a tree, with one vertex of degree k, and k vertices of degree 1.

$$GWD_\gamma(\mathbf{y}) = \sum_{d=0}^{N_v-1} e^{-\gamma d} N_d(\mathbf{y}) \ , \tag{6.32}$$

where $N_d(\mathbf{y})$ is the number of vertices of degree d and $\gamma > 0$ is related to λ through the expression $\gamma = \log[\lambda/(\lambda-1)]$. So this approach in a sense attempts to model the degree distribution, with choice of γ influencing the extent to which higher-degree vertices are likely to occur in the graph G.

Snijders, Pattison, Robins, and Handcock [360] discuss a number of other similar statistics, including a generalization of triadic structures based on alternating sums of k-triangles, which takes the form[21]

$$AKT_\lambda(\mathbf{y}) = 3T_1 + \sum_{k=2}^{N_v-2} (-1)^{k+1} \frac{T_k(\mathbf{y})}{\lambda^{k-1}} \ . \tag{6.33}$$

Here T_k is the number of k-triangles, where a k-triangle is defined to be a set of k individual triangles sharing a common base. These authors show that the inclusion of k-triangles in a model leads to a distribution $\mathbb{P}_\theta(\cdot)$ in which two potential edges are dependent, conditional upon the rest of the graph, if either (i) they share a common vertex, or (ii) their presence would create a four-cycle. As such, these models satisfy the so-called *partial conditional dependence* condition of Pattison and Robins [311], a non-Markov notion of dependence designed to allow more complex forms of inter-dependence than Markov dependence (e.g., such as through the status of 'third party' edges).

In fact, the ERGM framework is generally quite versatile and can easily allow for a number of additional generalizations. For example, directed versions of ERGMs are also available. The paper by Holland and Leinhardt [203] is seminal in this area, in moving from independent edges $Y_{ij} = Y_{ji}$ to independent dyads (Y_{ij}, Y_{ji}). The Markov models of Frank and Strauss [155] apply to this case as well, and similarly allow for a departure from this independence assumption. ERGMs also extend to bipartite graphs and multivariate networks (i.e., where more than one network is considered simultaneously on the same set of vertices). Finally, in defining ERGMs for either undirected or directed graphs, it is straightforward to include, if desired, additional information on vertices beyond their connectivity, such as actor attributes in a social network or known functionalities of proteins in a network of protein interactions. Given a realization $\mathbf{x}$ of a random vector $\mathbf{X}$ on the vertices in G, we simply specify an exponential form for the conditional distribution $\mathbb{P}_\theta(\mathbf{Y} = \mathbf{y} | \mathbf{X} = \mathbf{x})$ that involves additional statistics $g(\cdot)$ that are functions of both $\mathbf{y}$ and $\mathbf{x}$. An illustration may be found in Section 6.5.4.

[21] Hunter [208] offers an equivalent formulation of this definition, in terms of geometrically weighted counts of the neighbors common to adjacent vertices.

6.5.2 Fitting Exponential Random Graph Models

In standard settings, with independent and identically distributed realizations, exponential family models like that in (6.23) are generally fit using the method of maximum likelihood, and the resulting parameter estimates $\hat{\theta}$ are accompanied by asymptotically justified confidence intervals and test statistics. In the context of the ERGMs in (6.24), however, traditionally the corresponding inferential framework has been less well developed. The maximum likelihood estimators (MLEs) $\hat{\theta}_H$ of the parameters θ_H are well defined, but their calculation is non-trivial. Furthermore, an appropriate asymptotic theory for confidence intervals and testing, taking into account the highly dependent nature of observations in a network graph, has yet to be established. Research in this area has concentrated primarily on the problem of fitting MLEs for ERGMs; we review some of the key methods emerging from this work here.

Consider the general definition of an ERGM in (6.24). The MLE for the vector $\theta = (\theta_H)$ is defined as $\hat{\theta} = \arg\max_\theta \ell(\theta)$, where $\ell(\theta)$ is the log-likelihood, which has the particularly simple form common to exponential families,

$$\ell(\theta) = \theta^T \mathbf{g}(\mathbf{y}) - \psi(\theta) \ . \tag{6.34}$$

Here $\mathbf{g}$ denotes the vector of functions g_H and $\psi(\theta) = \log \kappa(\theta)$. Alternatively, taking derivatives on each side and using the fact that $\mathbb{E}_\theta[\mathbf{g}(\mathbf{Y})] = \partial\psi(\theta)/\partial\theta$, the MLE can also be expressed as the solution to the system of equations

$$\mathbb{E}_{\hat{\theta}}[\mathbf{g}(\mathbf{Y})] = \mathbf{g}(\mathbf{y}) \ . \tag{6.35}$$

Unfortunately, the function $\psi(\theta)$, occurring in both (6.34) and (6.35), cannot be evaluated explicitly in any but the most trivial of settings, as it involves the summation in (6.25) over $2^{\binom{N_v}{2}}$ possible choices of $\mathbf{y}$, for each candidate θ. Therefore, it is necessary to use numerical methods to compute approximate values for $\hat{\theta}$. Two Monte Carlo approaches are commonly used for this purpose, one based on the stochastic approximation of the log-likelihood in (6.34), and the other using a method for the stochastic approximation to solutions of systems of equations, applied to those in (6.35).

The first method, Markov chain Monte Carlo maximum likelihood estimation, derives from fundamental work of Geyer and Thompson [170]. Note that optimization of the log-likelihood in (6.34) is equivalent to optimization of the logarithm of the likelihood ratio

$$\begin{aligned} r(\theta, \theta^{(0)}) &= \ell(\theta) - \ell(\theta^{(0)}) \\ &= (\theta - \theta^{(0)})^T \mathbf{g}(\mathbf{y}) - \left(\psi(\theta) - \psi(\theta^{(0)})\right) \ , \end{aligned} \tag{6.36}$$

for arbitrary, fixed $\theta^{(0)}$. Furthermore, note that

$$\exp\left\{\psi(\theta)-\psi\left(\theta^{(0)}\right)\right\}=\sum_{\mathbf{y}}\exp\left\{\left(\theta-\theta^{(0)}\right)^{T}\mathbf{g}(\mathbf{y})\right\}\left(\frac{\exp\left\{\left(\theta^{(0)}\right)^{T}\mathbf{g}(\mathbf{y})\right\}}{\exp\left\{\psi\left(\theta^{(0)}\right)\right\}}\right)$$
$$=\mathbb{E}_{\theta^{(0)}}\left[\exp\left\{\left(\theta-\theta^{(0)}\right)^{T}\mathbf{g}(\mathbf{Y})\right\}\right]\ . \tag{6.37}$$

Therefore, an approximation to the term $\psi(\theta)-\psi(\theta^{(0)})$ in (6.36) can be produced by (i) generating a Markov chain Monte Carlo sample $\mathbf{Y}_1,\ldots,\mathbf{Y}_n$ from the ERGM (6.24), under $\theta^{(0)}$, (ii) approximating the expectation in (6.37) by the corresponding average based on this sample, and (iii) taking a logarithm of that average. The resulting approximation to the log-likelihood ratio will converge to its target as n tends to infinity, and hence the optimum of this approximate log-likelihood ratio will approximate the MLE $\hat{\theta}$. See Hunter and Handcock [209], for example, for additional details and references.

The second method utilizes the algorithm of Robbins and Monro [324], which may be viewed as a stochastic version of the Newton-Raphson algorithm. Given a random vector $\mathbf{Z}$, with distribution parameterized by a vector θ, the Robbins-Monro algorithm allows for the solution in θ of the system of equations $\mathbb{E}_\theta[\mathbf{Z}]=0$, through a sequence of iterations of the form

$$\hat{\theta}^{(i+1)}=\hat{\theta}^{(i)}-a_i D_i^{-1}\mathbf{Z}_i\ . \tag{6.38}$$

Here the $\mathbf{Z}_i$ are a sequence of random vectors for which the distribution of $\mathbf{Z}_i$, conditional on $\mathbf{Z}_1,\ldots,\mathbf{Z}_{i-1}$, is that of $\mathbf{Z}$ at $\theta=\hat{\theta}^{(i)}$; the a_i are a sequence of positive numbers tending to zero; and D_i is a matrix playing the role of the Hessian in the traditional Newton-Raphson algorithm. In the case of $a_i=1/i$ and $\mathbf{Z}$ distributed according to an exponential family, as in (6.23), the optimal choice of D_i is $\mathsf{Cov}_\theta(\mathbf{Z})$, although in practice this quantity is usually approximated. For the task of estimating the MLE $\hat{\theta}$ that solves the system of equations in (6.35) deriving from the ERGM in (6.24), we set $\mathbf{Z}=\mathbf{g}(\mathbf{Y})-\mathbf{g}(\mathbf{y})$, where $\mathbf{Y}$ is distributed according to (6.24). See Snijders [357] for additional details.

Note that for both of these methods of approximating the MLE, it is necessary to be able to simulate draws of $\mathbf{Y}$ from the ERGM in (6.24). One natural approach to this task is to use the Gibbs sampler, a general Markov chain Monte Carlo method for simulating from the joint distribution of a vector $\mathbf{Z}$ that utilizes only the univariate conditional distributions of each element given all of the others.[22] For ERGMs, such conditional distributions have a particularly simple form. Writing $\mathbf{Y}_{(-ij)}$ to be all of the elements of $\mathbf{Y}$ except Y_{ij}, the distribution of Y_{ij} conditional on $\mathbf{Y}_{(-ij)}$ is Bernoulli and satisfies the expression

$$\log\left[\frac{\mathbb{P}_\theta(Y_{ij}=1|\mathbf{Y}_{(-ij)}=\mathbf{y}_{(-ij)})}{\mathbb{P}_\theta(Y_{ij}=0|\mathbf{Y}_{(-ij)}=\mathbf{y}_{(-ij)})}\right]=\theta^T\Delta_{ij}(\mathbf{y})\ , \tag{6.39}$$

[22] We describe Gibbs sampling in somewhat greater detail in Section 8.3.2.2.

where $\Delta_{ij}(\mathbf{y})$ is the *change statistic*, denoting the difference between $\mathbf{g}(\mathbf{y})$ when $y_{ij} = 1$ and when $y_{ij} = 0$, which may be calculated in an efficient manner.[23] Various other MCMC methods are also possible. See Snijders [357] for some discussion.

A disadvantage of both of the methods above is their computationally intensive nature. To date they have been applied to networks with at most a few thousand vertices. An alternative is to estimate θ by maximizing not the actual log-likelihood (6.34), but rather the *log-pseudo-likelihood*

$$\sum_{\{i,j\}} \log \mathbb{P}_\theta(Y_{ij} = 1 | \mathbf{Y}_{(-ij)} = \mathbf{y}_{(-ij)}) \ . \tag{6.40}$$

This approach, originally proposed by Besag [37] in the context of spatial data analysis, and adapted for ERGMs by Strauss and Ikeda [372], will work best (i.e., in the sense of producing an estimate that reasonably approximates the MLE $\hat{\theta}$) when dependencies among the elements of $\mathbf{Y}$ are relatively weak. Unfortunately, in many network contexts this is not likely to be the case. Nevertheless, pseudo-likelihood estimation is a computationally expedient[24] method for obtaining at least some rough sense of the value of $\hat{\theta}$. Historically, this approach has in fact been the primary method of estimation, until the comparatively more recent development of reliable MCMC methods. See Robins et al. [329] for a comparison of results from pseudo-likelihood and MCMC methods, when applied to some commonly studied social networks.

6.5.3 Goodness-of-Fit and Model Degeneracy

In any sort of modeling problem, the best fit chosen from among a class of models need not necessarily be a *good* fit to the data if the model class itself does not contain a sufficiently rich set of models from which to choose. The concept of model *goodness-of-fit* is therefore important. But, while this concept is fairly well developed in standard modeling contexts, such as linear modeling, it is arguably still in its infancy as far as network graph modeling is concerned.

For ERGMs, the current practice in assessing goodness-of-fit is to first simulate numerous random graphs from the fitted model and then compare high-level characteristics of these graphs with those of the originally observed graph. Examples of such characteristics include the distribution of any number of the various summaries of network structure encountered in Chapter 4, such as degree, centrality, and geodesic distance. If the characteristics of the observed network graph are too poor of a match to the typical values arising from realizations of the fitted random graph model (e.g., as viewed by appropriate plots, such as those shown in Section 6.5.4),

[23] In particular, its calculation does not involve $\psi(\theta)$.

[24] Maximum pseudo-likelihood estimation here is equivalent to logistic regression of the elements of $\mathbf{y}$ on the design matrix formed by the vectors $\Delta_{ij}(\mathbf{y})$, and may thus be carried out using standard software.

then this suggests systematic differences between the specified class of models and the data, and therefore a lack of goodness-of-fit.[25]

Goodness-of-fit has been found to be particularly important where ERGMs are concerned, due in large part to the issue of *model degeneracy*. In this context the term is used to refer to a probability distribution that places a disproportionately large amount of its mass on a correspondingly small set of outcomes. See Handcock [193]. A number of simple but originally popular Markov graph models have been shown to be degenerate. A common case is where the ERGM places most of its mass on either the empty graph, the complete graph, or a mixture of the two, depending on the value of θ. None of these, of course, is likely to be especially appropriate for modeling real data of any interest.

In addition to its defining lack of richness, model degeneracy also can lead to difficulties in fitting ERGMs. This is due to the fact that, for degenerate models, the transitions among the limited outcomes (e.g., from null, to mixture, to complete), as θ changes, can be quite sharp. As a result, for poorly specified ERGMs, numerical algorithms like MCMC can have substantial difficulty converging. See Handcock [193], for example, for a careful study of the relationship between model degeneracy and model fitting.

In summary, model specification for ERGMs is clearly a nontrivial task and one that should be approached in a manner closely informed by the above issues, with goodness-of-fit diagnostics playing an important facilitating role. These issues were in fact a major motivation for the proposal of statistics like the alternating k-stars and alternating k-triangles mentioned above. See Snijders, Pattison, Robins, and Handcock [360] for an extensive discussion in this context, including examples and further references.

6.5.4 Case Study: Modeling Collaboration Among Lawyers

In this case study we illustrate the application of exponential random graph models using a dataset on collaborative working relationships among members of a New England law firm, collected by Lazega [251]. Our analysis draws upon original analyses in Hunter and Handcock [209] and Snijders, Pattison, Robins, and Handcock [360].

Lazega's data were collected for the purpose of studying cooperation among social actors in an organization, through the exchange of various types of resources among them. The organization observed was a corporate law firm, consisting of over 70 lawyers (roughly half partners and the other half associates) in three offices located in three different cities. Relational data reflecting resource exchange were collected, and additional attribute information was recorded for each lawyer. See Lazega and Pattison [252] for additional details.

[25] This simulation-based approach to evaluating models is actually used more broadly, such as when parameters are simply 'hand-tuned' rather than fitted by formal statistical methods.

Here we use only a portion of Lazega's dataset. First, we restrict our attention to only the $N_v = 36$ partners in the firm, among whom the administrative structure was said to be 'relatively decentralized.' Second, we focus upon the network of collaboration among partners, where an edge is present in our network graph if and only if both partners indicated in a survey that they worked together in a substantive manner. Third, we include data on seniority (i.e., the rank entry of the partner's entrance into the law firm), gender, office location, and type of practice (i.e., litigation or corporate law). Figure 6.7 shows a visual representation of this data.

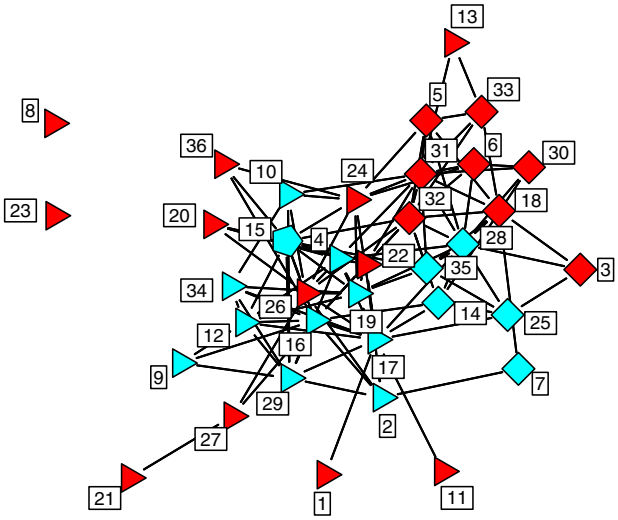

Fig. 6.7 Visualization of Lazega's network of collaborative working relationships among lawyers. Vertices represent partners and are labeled according to their seniority. Vertex shapes (i.e., triangle, square, or pentagon) indicate three different office locations, while vertex colors correspond to the type of practice (i.e., litigation (red) or corporate (cyan)). Edges indicate collaboration between partners. There are three female partners (i.e., those with seniority labels 27, 29, and 34); the rest are male. Data courtesy of Emmanuel Lazega.

It is of interest in a study like this to assess the effects on collaboration of the exogenous attribute variables, while controlling for certain endogenous effects of network structure that might be anticipated, and vice versa. Towards this end, we specify an ERGM like that in (6.24) with two network statistics and five statistics involving attributes. Drawing on the analyses in Hunter and Handcock [209] and Snijders, Pattison, Robins, and Handcock [360] to discard uninformative statistics, the network statistics we chose to incorporate were the number of edges $S_1(\mathbf{y}) = N_e$ and the alternating k-triangles statistic $AKT_\lambda(\mathbf{y})$ defined in (6.33). These allow us

to control for the density of the network and some effects of transitivity. Following Snijders, Pattison, Robins, and Handcock [360], we use a value of $\lambda = 3.0$ in computing the AKT_λ statistic.[26]

The five variables relating to attributes all are of the form

$$g(\mathbf{y}, \mathbf{x}) = \sum_{1 \le i < j \le N_v} y_{ij} h(\mathbf{x}_i, \mathbf{x}_j) \ , \tag{6.41}$$

where h is a symmetric function of $\mathbf{x}_i$ and $\mathbf{x}_j$, and $\mathbf{x}_i$ (or $\mathbf{x}_j$) is the vector of observed attributes for the i-th (or j-th) vertex. Our choice of h falls into two categories, producing 'main effects' and 'second-order effects' (or similarity or homophily effects) of certain attributes. Main effects are defined for the seniority and practice attributes using a simple additive form:

$$h(\mathbf{x}_i, \mathbf{x}_j) = \text{seniority}_i + \text{seniority}_j \tag{6.42}$$

for seniority and analogously for practice. Second-order effects are defined for practice, gender, and office location, and in each case h is simply an indicator for equivalence of the respective attribute between two vertices, such as $h(\mathbf{x}_i, \mathbf{x}_j) = I\{\text{gender}_i = \text{gender}_j\}$.

The overall model is therefore of the form

$$\mathbb{P}_{\theta,\beta}(\mathbf{Y} = \mathbf{y} | \mathbf{X} = \mathbf{x}) = \left(\frac{1}{\kappa(\theta, \beta)} \right) \exp\left\{ \theta_1 S_1(\mathbf{y}) + \theta_2 AKT_\lambda(\mathbf{y}) + \beta^T \mathbf{g}(\mathbf{y}, \mathbf{x}) \right\} \ , \tag{6.43}$$

where $\mathbf{g}$ is the vector of five attribute statistics and β is the corresponding vector of parameters. The model was fit using the `statnet` package,[27] which implements the MCMC maximum likelihood procedure described in Section 6.5.2, and in addition contains various features to facilitate monitoring of convergence of the underlying Markov chains.

Table 6.1 shows the fitted values for the parameters in our model and an accompanying set of 'standard errors.' These standard error values are obtained by applying classical asymptotic theory, which, as was already discussed in Section 6.5.2, is strictly speaking unjustified in the face of the dependencies in our data. As a result, such values and any corresponding statements of significance are generally to be taken as heuristics. Here we see that the evidence of the regression suggests that all effects are potentially 'significant,' in the sense of having a non-trivial ability to

[26] Alternatively, we might consider treating λ in statistics like (6.31) and (6.33) as an additional parameter in the model, to be estimated from the data. But in this case, the model $\mathbb{P}_\theta(\mathbf{Y} = \mathbf{y})$ no longer satisfies the exponential family form in (6.23). Rather, it is a *curved exponential family.* Hunter and Handcock [209] present a modified form of the MCMC maximum likelihood algorithm described in Section 6.5.2 to fit such network graph models. For the analysis of this particular dataset, they report an estimate $\hat{\lambda}$ that does not appear to differ in a statistically significant manner from the choice $\lambda = 3.0$. However, this is not entirely surprising, given that this λ was chosen in Snijders, Pattison, Robins, and Handcock [360] after comparing results for several different values.

[27] Available at *http://csde.washington.edu/statnet*.

Parameter	Estimate	'Standard Error'
Density (θ_1)	-6.2073	0.5697
Alternating k-triangles (θ_2)	0.5909	0.0882
Seniority Main Effect (β_1)	0.0245	0.0064
Practice Main Effect (β_2)	0.3945	0.1103
Same Practice (β_3)	0.7721	0.1973
Same Gender (β_4)	0.7302	0.2495
Same Office (β_5)	1.1614	0.1952

Table 6.1 Results from fit of model (6.43) to the Lazega lawyer data.

explain some of the variation in network structure, since the ratio of each estimate to its standard error is on the order of 3 or more.

Recalling the expression in (6.39), note that the estimated coefficient of each attribute statistic may be interpreted as a conditional log-odds ratio for cooperation. For example, practicing corporate law, rather than litigation, increases the odds of cooperation by a factor of $\exp(0.3945) \approx 1.484$, or nearly 50%. Similarly, being of the same gender more than doubles the odds of cooperation, since $\exp(0.7302) \approx 2.076$. In all cases, such statements hold in the sense of 'all else being equal' (i.e., given no change among values of the other statistics).

In terms of network structure, the magnitude of the coefficient $\hat{\theta}_2 \approx 0.5909$ for the alternating k-triangle statistic and the comparatively quite small corresponding standard error indicate that there is strong evidence for a substantial transitivity effect. Note that, given the inclusion of our second-order attribute statistics in the model, our quantification of this effect naturally controls for basic homophily on these attributes. So there is likely something other than similarity of gender, practice, and office at work here – possibly additional attributes we have not controlled for, or possibly social processes of team formation.

In order to assess the goodness-of-fit of our model, we simulated realizations from the model (6.43) with the parameters in Table 6.1. The structure of the resulting random network graphs and the original network graph were compared in terms of the distribution of degree, geodesic length, and edge-wise shared partners (i.e., the number of neighbors shared by a pair of vertices defining an edge). The results are shown in Figure 6.8, and indicate that – on these particular characteristics – the fit of the model is quite good overall.

6.6 Challenges in Modeling Network Graphs

In designing network graph models, it is crucial to maintain sight of the fact that the network topology structure arises in conjunction with the workings of an underlying complex system. A fundamental challenge in this area is to adequately incorporate our understanding of such workings into network graph models that are nevertheless still at least computationally, if not also analytically, tractable.

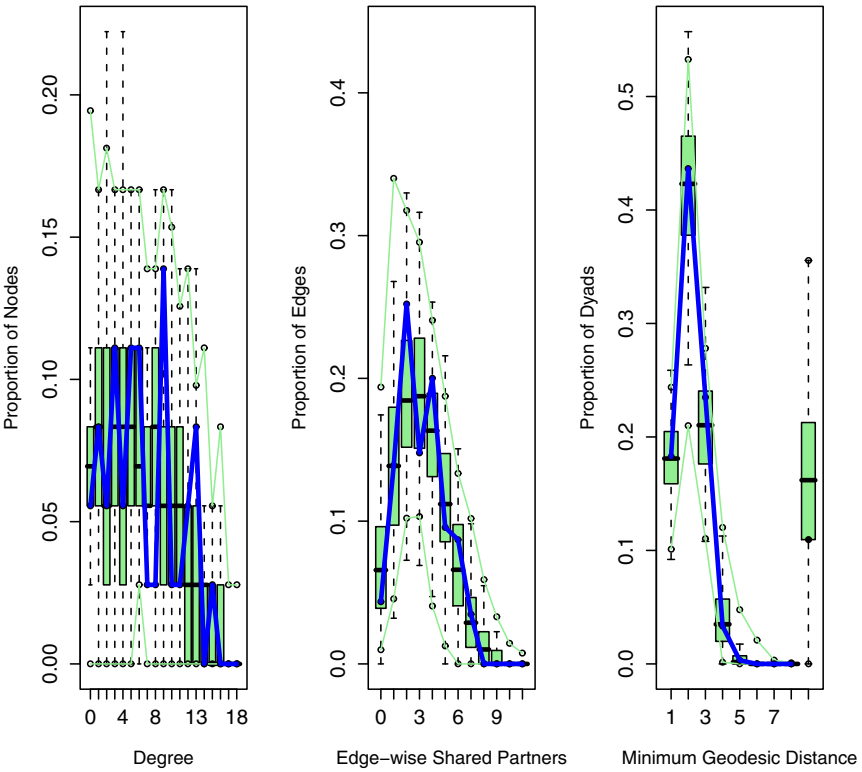

Fig. 6.8 Goodness-of-fit plots comparing original Lazega lawyer network and 100 realizations from the model in (6.43), with the parameters in Table 6.1. Comparisons are made based on the distribution of degree, edge-wise shared partners, and geodesic distance over the 100 realizations, represented by box-plots and curves showing 10-th and 90-th quantiles – both in green. Values for the Lazega network itself are shown with solid blue lines. In the distribution of geodesic distances between pairs, the rightmost box-plot is separate and corresponds to the proportion of nonreachable pairs.

Preferential attachment and copying, for example, are two simple, generic mechanisms that have been somewhat successful in reproducing certain gross structural characteristics of the degree distribution in various real-world networks. Nevertheless, as a result of their generic nature, they arguably capture the internal workings of the systems underlying these networks in only the crudest of manners. Work in this area continues to try to refine mechanisms of this sort.

Similarly, ERGMs too allow for the inclusion of effects of certain simple mechanisms, in their own way, such as transitivity, but they also can incorporate the effects of potentially important vertex attribute information. And, notably, they allow for

some assessment of the strength of association between the observed network topology and the proposed mechanisms and attributes. However, there remains much to be done in this area, especially with respect to model specification and inference.

Alternatively, in contrast to the modeling approaches discussed here, Li, Alderson, Willinger, and Doyle [256] have advocated a more holistic approach to network graph modeling, in which network growth is seen as influenced by various types of constraints and trade-offs. Their 'first principles' approach to constructing network graphs, illustrated in the context of modeling the router-level Internet (i.e., the Internet as described in Section 3.5.2), uses a blend of standard statistical data analysis tools and graph theory and, most importantly, various technology constraints and economic factors, to generate networks with topologies much more similar to those observed than produced by other models. However, their overall modeling approach – *heuristically optimal topology* (HOT) – a heuristically driven, non-random rewiring algorithm, necessarily must be taken more as an important demonstration and proof of concept than a formal methodology.

A dynamical systems perspective, which we have not touched upon here in this chapter, has – in principle – much to offer in this direction. The complex systems studied using network methods usually are indeed dynamic and a network graph G frequently is just a snap-shot of system behavior, either at a given point in time or perhaps cumulatively. But the sophisticated use of stochastic dynamical systems models for network formation, particularly in contexts with networks of non-trivial size, is still arguably in its infancy. There remain numerous challenges to be faced, ranging from improved understanding of the dynamics of such models, to how to fit them, to obtaining sufficient data to do so. We will return briefly to this topic area in Chapter 8, when we explore the related issue of modeling dynamic processes indexed on network graphs, in which the graph is assumed fixed and known.

One related area in which some progress has been made is that of agent-based modeling, which has its roots in the intersection of the more quantitative branches of economics, psychology, and sociology. In this approach, a game-theoretic perspective drives the study of network formation, with a particular eye towards how individual incentives balance with the global 'greater good' when agents (e.g., social actors) are allowed choice in establishing relationships. See the survey by Jackson [212], for example.

Developments in these various areas of network graph modeling increasingly continue to inform each other, as researchers in these diverse fields interact more and more. Presumably it is in such cross-fertilization that there lies the greatest potential for breakthroughs.

6.7 Additional Related Topics and Reading

There are a number of books devoted entirely to subsets of the topics in network graph modeling discussed in this chapter. The volume by Bollobás [43] is the standard reference for classical random graph theory. The book of Chung and Lu [88]

contains a unified exposition of numerous formal results on generalized random graph, network growth, and small-world models. The book by Dorogovstev and Mendes [123] presents a physics perspective on network growth (a.k.a. 'evolutionary') models.

A number of general literature surveys have been written on network graph modeling. Two that we have found particularly useful are the reviews by Newman [296] and by Boccaletti et al. [41]. Both are written from the perspective of the statistical mechanics of complex networks, but with a fairly general audience in mind. For an excellent overview and illustration of ERGMs, with an emphasis on more recent advances, see the special section in the May 2007 issue of the journal *Social Networks.*

While we have attempted to do reasonable justice to the topic of network graph modeling in this chapter – in a not unreasonable amount of space – by concentrating on what to date have been the main model classes of interest, there are of course other classes that call for attention as well. One example is the class of *weighted networks*, wherein the strength of the connection between vertices (i.e., the edge weight) is included in the model. See Boccaletti et al. [41, Sec. 2.4.3] for a short survey of work in this area. Another example is the class of latent variable models, as developed by Nowicki and Snijders [304], Hoff [199], and Hoff, Raftery, and Handcock [202], which augment the basic Bernoulli model in (6.26) with a second layer of modeling for the parameters θ_{ij}. We will see an example of such models in Chapter 7 in the context of predicting edges missing from a partially observed graph. Also, there are dynamic extensions of ERGMs and related models, in which a sequence of observed network graphs is modeled, for example through the addition of a Markov component in time. See Snijders [358], for example, for a review, and also Koskinen and Snijders [236].

Finally, on the topic of ERGMs, being as they are probabilistic models with conditional dependency relationships defined in terms of graph structure, we would be remiss not to point out here that they are, in fact, one example of the more general class of models known as *graphical models.* Technical tools like the Hammersley-Clifford theorem cited earlier in this chapter are fundamental to this area. We will encounter a handful of additional specific instances of graphical models over the course of the next three chapters, and in Chapter 10 we will then look briefly at the general topic of graphical models in its own right.

Exercises

6.1. Recall the problem of model-based estimation of the size of a 'hidden population,' as described in Example 6.3.

a. Verify the expressions for the expectations in (6.4), (6.5), and (6.6).

b. Show that the corresponding method-of-moments equations

$$\begin{aligned} n &= N_v\, p_0 \\ m_1 &= N_v\,(N_v-1)\, p_0^2\, p_{\mathscr{G}} \\ m_2 &= N_v\,(N_v-1)\, p_0\,(1-p_0)\, p_{\mathscr{G}} \end{aligned}$$

yield the estimates $\hat{p}_0$, $\hat{p}_{\mathscr{G}}$, and $\hat{N}_v$ of p_0, $p_{\mathscr{G}}$, and N_v given in equations (6.7), (6.8), and (6.9), respectively.

6.2. A semi-rigorous proof of the limiting power-law behavior of the degree distribution for the Barabási-Albert preferential attachment model rests upon the use of so-called 'rate equations.'

a. By considering the expected change in the number of vertices of degree d, from iteration t to iteration $t+1$, argue that for $d > m$,

$$(t+1)\, f_d^{(t+1)} - t\, f_d^{(t)} = \frac{(d-1)\, f_{d-1}^{(t)}}{2} - \frac{d\, f_d^{(t)}}{2} \quad ,$$

and for $d = m$,

$$(t+1)\, f_m^{(t+1)} = 1 - \frac{m\, f_m^{(t)}}{2} \quad .$$

Here $f_d^{(t)} \equiv f_d\left(G^{(t)}\right)$.

b. Under the assumption that $f_d^{(t)}$ tends to the value f_d (i.e., $\lim_{t\to\infty} f_d^{(t)} = f_d$), for each d, with respect to some limiting distribution $\{f_d\}$, use part (a) to argue that

$$f_d = \begin{cases} \frac{1}{2}(d-1)\, f_{d-1} - \frac{1}{2}\, d\, f_d, & \text{if } d > m, \\ 1 - \frac{1}{2} m\, f_m, & \text{if } d = m \ . \end{cases}$$

c. Use this result to conclude that $f_d = f_{d,m}$ is given by the expression in (6.17).

6.3. We consider here the problem of estimating the exponent of a power-law degree distribution, as described in Section 4.2.1.1.

a. Implement one of the network growth models described in Section 6.4, so as to be able to simulate realizations of your model using a computer. For example, the LCD preferential attachment model can be implemented using the algorithm of

Batagelj and Brandes [23]. For various choices of t, compare the average degree distribution obtained with the distribution of the limiting power law.

b. Explore the accuracy with which you can estimate the power-law exponent from a single realization of your model, for sufficiently large t, using the Hill estimator (4.4). Compare the performance of this estimator with that of the naive regression estimator motivated by the relation in (4.2) and the logarithm of the relation in (4.3).

6.4. Recall the important role that the log-odds ratio plays in the fitting and interpretation of ERGMs.

a. For the exponential random graph model in (6.24), verify that the log-odds ratio

$$\log\left[\frac{\mathbb{P}_\theta(Y_{ij}=1|\mathbf{Y}_{(-ij)}=\mathbf{y}_{(-ij)})}{\mathbb{P}_\theta(Y_{ij}=0|\mathbf{Y}_{(-ij)}=\mathbf{y}_{(-ij)})}\right]=\theta^T\Delta_{ij}(\mathbf{y}) \ ,$$

as stated in equation (6.39), where $\Delta_{ij}(\mathbf{y})$ is the change statistic.

b. In the case of the Markov graph model (6.30), with terms $S_1(\mathbf{y})$, $S_2(\mathbf{y})$, and $T(\mathbf{y})$ in the model, show that the change statistic is given by

$$\Delta_{ij}(\mathbf{y})=\begin{pmatrix}1\\ y_{i+}+y_{j+}-2y_{ij}\\ \sum_{k\neq i,j}y_{ik}y_{kj}\end{pmatrix} \ .$$

6.5. We saw two methods – in Example 6.4 and in the case study of Section 6.5.4 – for assessing the evidence for transitivity in a given network while controlling for certain other factors. For a network of your choice, and using appropriate software, assess the evidence for transitivity in your network using one or both of these methods. Report on your findings and discuss any implications of these findings on the system underlying your network.