Blockmodeling techniques for complex networks

by

Brian Joseph Ball

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Physics)
in The University of Michigan
2014

Doctoral Committee:

Professor Mark E. Newman, Chair
Professor Charles R. Doering
Associate Professor Elizaveta Levina
Assistant Professor Xiaoming Mao
Professor Michal R. Zochowski

For my parents

ACKNOWLEDGEMENTS

I would like to thank everyone who has helped make this dissertation a reality, starting with my advisor, Dr. Mark Newman. Mark's extraordinary passion and curiosity have been an inspiration to me. His vast knowledge and patience have been invaluable to my education and professional development. I would also like to thank Dr. Charles Doering, Dr. Elizaveta Levina, Dr. Xiaoming Mao, and Dr. Michal Zochowski, who have kindly taken the time to serve on my dissertation committee.

I would also like to thank Dr. Jean Krisch, who has helped mentor me and guide my career since before I came to Michigan as a freshman. Her wisdom and genuine concern for my success have been a strong positive influence throughout my time here.

Special thanks go to all of my officemates in 320 West Hall, including Brian Karrer, Travis Martin, and Thomas Zhang, who somehow managed to put up with me on a daily basis. It was a pleasure working with such intelligent people with whom I could work and share ideas. My cohort members Jax Sanders and Adam Sypniewski also deserve thanks for their friendship and fresh perspectives in many discussions. I would also like to thank all of my friends from my church's graduate student and young professional group. I would never have made it this far without their encouragement and support.

Finally, I give my biggest thanks to my parents, to whom I owe everything.

This work uses data from Add Health, a program project designed by J. Richard Udry, Peter S. Bearman, and Kathleen Mullan Harris, and funded by a grant P01–

HD31921 from the National Institute of Child Health and Human Development, with cooperative funding from 17 other agencies. Special acknowledgment is due Ronald R. Rindfuss and Barbara Entwisle for assistance in the original design. Persons interested in obtaining data files from Add Health should contact Add Health, Carolina Population Center, 123 W. Franklin Street, Chapel Hill, NC 27516–2524 (addhealth@unc.edu).

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

Figure

LIST OF TABLES

Table

LIST OF APPENDICES

Appendix

CHAPTER I

Introduction

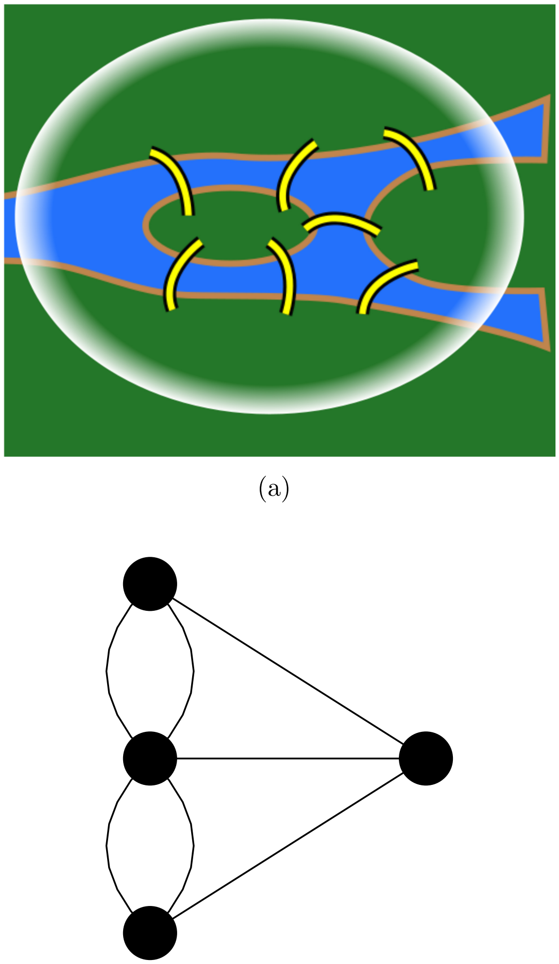
The word “network” has become pervasive in our society. The average person understands that networks are found almost everywhere and will immediately produce such examples as the internet or social networks like Facebook. Pressed a little harder, people can come up with other examples, like the power grid, food webs, and cell phone networks. What most people do not realize however, is that while these networks are different, there is a single set of mathematics that scientists use to describe all of them that unifies them together as networks. This combined understanding is the result of discovering a common thread through years of work from scientists in unrelated fields studying seemingly unrelated topics; each of the examples above and in fact all networks stem from the same building blocks, a set of objects (vertices) and connections between them (edges). The result of this perspective is a powerful tool with far-ranging applications.

The first chapter of this dissertation begins by providing background into the subject of networks. Section 1.1 discusses the many histories of networks and how they came together. This leads to section 1.2, which talks about why physicists are interested in networks, their main contributions to the field, and how they are necessary for the continued success in the modern world of networks. Section 1.3 discusses all of the network basics relevant to the dissertation. This introductory chapter wraps up in section 1.4 with the novel contributions of this dissertation to network science and a summary of the main chapters of the dissertation.

1.1 Historical interest

The first interest in networks developed from the mathematical study of graph theory. One of the best known early problems that sparked interest in the field came from Leonhard Euler's 1735 study of bridges in the Prussian city of Königsberg. The problem posed was this: how could a person cross each of the seven bridges in the city, situated over the Pregel River, without crossing the same bridge twice? This problem is demonstrated in figure 1.1a. Euler's genius was to acknowledge that the shapes of the landmasses were irrelevant – all that mattered was which landmasses were connected by the bridges. The entire problem can be reduced to traversing edges on a graph, shown in figure 1.1b (not surprisingly, this problem is now in general referred to as finding an Eulerian path), and Euler showed that in the case of the Königsberg bridges, there is no solution.

While graph theory has been around since Euler's time, its primary focus has been purely mathematical – there has been very little application to real-world networks. Sociologists independently started gaining interest in networks in the early 20th century. An important example is Jacob Moreno's 1934 study of school children on the playground [98]. He drew a picture of the children and their interactions, shown in figure 1.2, which he termed a "sociogram", where the students were represented as triangles and circles for boys and girls respectively and a line was drawn between two students if they were seen interacting on the playground. From this picture, it was easy to see patterns in the interactions, specifically that the boys and girls primarily interacted within their own groups. Sociologists quickly saw the use of such pictures and modes of thinking and have been applying and studying social networks ever since. Of particular note are two specific models, the blockmodel and a tiered



(a)

(b)

Figure 1.1: (a) The Königsberg bridge problem. The goal is to cross each of the yellow bridges exactly once. (b) The picture rewritten as a network where the vertices are the land and the edges are the bridges. No Eulerian path exists in this network.

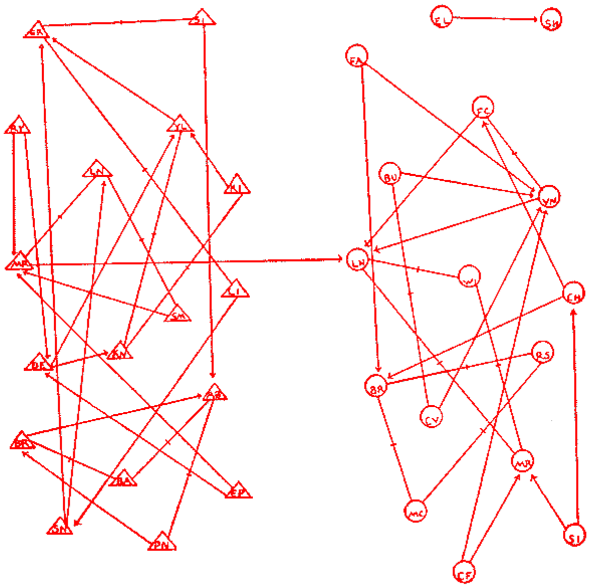


Figure 1.2: Jacob Moreno's sociogram [98]. The triangles are boys and the circles are girls. There are many interactions between boys and between girls, but only a single case where a boy interacted with a girl.

ranking model. In the blockmodel, the ideal situation is that everyone in a particular social clique connects to everyone else within the clique and connects to no one else[1]. Homans' tiered directed model was based upon an idea of social status where a person is allowed to connect to other people at their same level as well as make connections up the tiers, but is prohibited from connecting to a lower level [35, 75].

The two disciplines listed above are the forerunners of networks as a field. Many different scientific disciplines have contributed to the field, too numerous to discuss here, except to mention the contributions from computer scientists and biologists. One of the questions that attracted computer scientists' interest in the field was a problem mathematicians had studied for years, the traveling salesman. In this problem, a salesman attempts to travel between several cities. The roads between each

[1] In fact, the term "clique" is still used in networks to describe a set of fully connected vertices.

city have an associated cost of time and effort, and the salesman's goal is to minimize the total cost. This and several other problems united computer scientists and their computational complexity theory with networks [81]. The work done was especially relevant in a practical sense given the parallel development of local computer networks and the much larger connecting network, now known as the internet. Vulnerabilities in this new physical network were a real issue – scientists needed to be certain that any two computers could communicate with each other even as some connections failed, and that making the connection would be efficient with respect to the network. The main contributions of computer scientists to the early networks literature were minimum cut/maximum flow algorithms and minimum spanning trees (via path distance) in the form of local routing approximation algorithms [40, 53, 93]. A different topic that should also receive mention are computer scientists' work in classification with an emphasis on natural language and image processing that has applications to networks [19, 68, 69, 70, 88, 121].

Biologists on the other hand became interested in food webs and more recently metabolic and neural networks [31, 147]. Every ecosystem has a network where organisms with a higher so-called trophic level eat organisms with lower trophic levels, creating a chain connecting the lowest trophic levels up to the highest ones. These networks tell biologists about the stability of an ecosystem and thus are of central interest to ecologists. Similarly metabolic and neural networks are central to understanding how a particular organism works, so they have also received a lot of attention.

Since the early 1990s, the unification of all these different representations and applications became the field of network theory, with the review by Wasserman and Faust playing a significant part [143], and is largely due in part to the involvement of physicists and computer scientists.

1.2 Interest to physicists

Traditionally, networks were small systems of interacting agents [75, 85, 152]. Researchers studying the systems were generally interested in small scale features, on the order of individual node properties. For example, a scientist might ask how important a particular vertex is to the network, often answered in the form of centrality or connectivity measures [143]. However, with the advent of large scale computing and tools such as the internet, it became significantly easier to gather a large data set. Up to that point, a "large" network would be one with one or a few hundred agents. It was now possible to have data sets numbering in the millions and billions of agents. Rather than having specialized knowledge about each actor and interaction in the network, researchers could now only afford to have global knowledge about the entire system. Questions that had been central to networks in the past became difficult to interpret because the amount of information about individual agents got to be too large to handle. Even visual checks for interesting features are difficult, as can be demonstrated by comparing Moreno's sociogram and the internet[2], in figures 1.2 and 1.3 respectively.

Physicists are already well-acquainted with the problems associated with large datasets. On the experimental side, high energy physics experiments are, as of 2013, able to gather terabytes of data each day even after throwing away a significant fraction of the data. Dealing with large amounts of data can be a technical challenge, and physicists are better acquainted than most with this problem. From a theoretical standpoint, large systems are the routine study of statistical physics, which focuses on extremely large numbers of interacting particles. An early use of statistical methods in physics was the application of statistical mechanics to thermodynamics, which deals with physical properties of materials and fluids. One research topic in

[2] Technically this is not a full representation of the internet, but rather a subset of the edges defined by a minimum distance spanning tree as measured by route tracing from a single source computer [23].

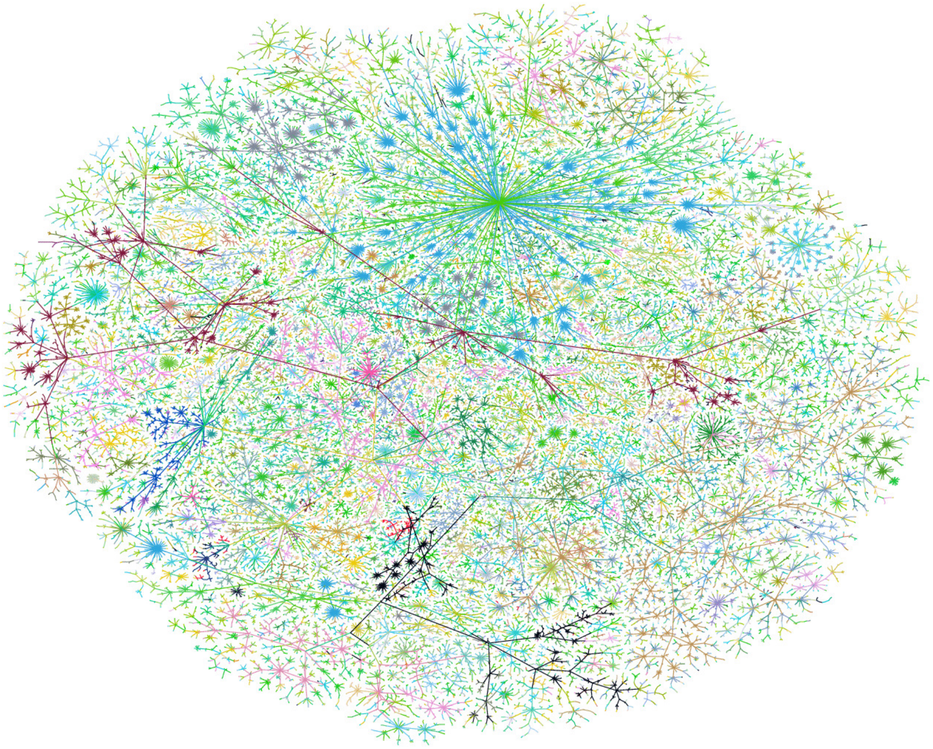
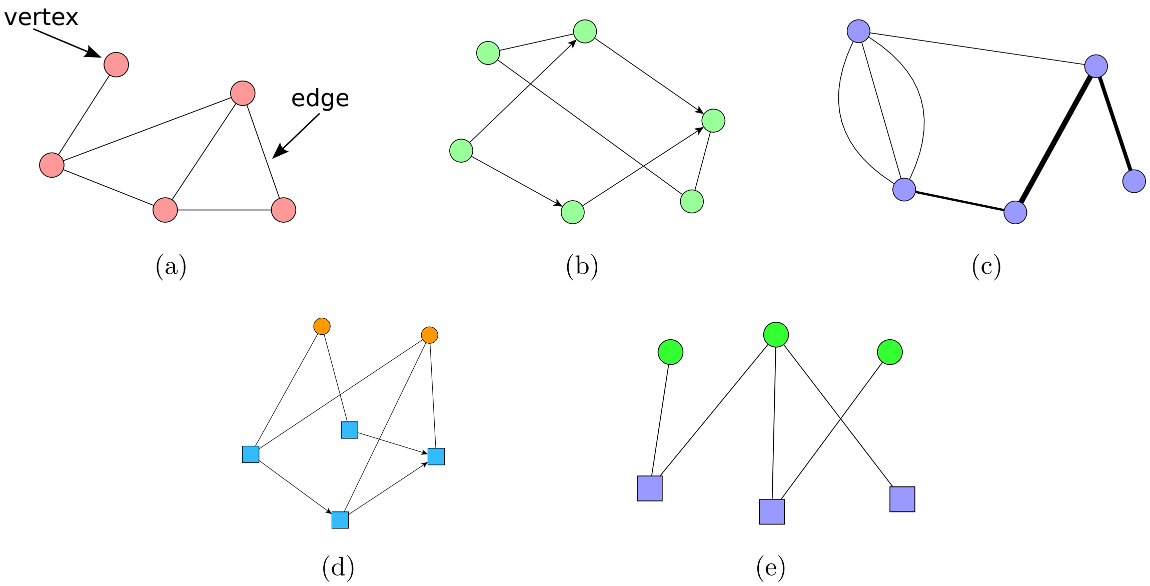


Figure 1.3: Visualization of the internet taken on June 29, 1999, colored by IP address. Credit for the image goes to Hal Burch and Bill Cheswick of the Internet Mapping Project [23].

statistical physics of particular interest to this dissertation is the interaction of particle spin systems in crystals, famously receiving attention from Ernst Ising in the 1920s [78, 116]. This is an important physics topic since spins affect the electromagnetic properties of a material. Crystals are defined as having a regular, strongly defined pattern of connections known as a lattice. In that regard, they can be viewed as being well-behaved networks. Between the wide availability of data from an abundance of sources (including but not limited to the examples given in section 1.1) and already being interested in similar problems, the expansion of physicists' interests to include studying networks was a natural one.

1.3 Basics of network structure

In general, a network is a collection of objects, known as vertices, nodes, or actors, and connections (typically pairwise) between them, called edges, as demonstrated in figure 1.4a [106]. There are many different ways these can manifest. For example, edges can be either directed or undirected (fig. 1.4b). There can also be multiple edges between the same pair of vertices, known as a multi-edge, or they can potentially have weights on them (fig. 1.4c). A network with no multi-edges or self-edges (an edge from a vertex to itself) is called simple. It is also possible to have multiple kinds of vertices and edges (fig. 1.4d). A special case of these multi-modal networks are bipartite networks, where there are two kinds of vertices that are allowed to connect to each other but not to vertices of their own type (fig. 1.4e). Every bit of understanding that can be achieved with networks builds off these concepts, so this section will cover the mathematics associated with these concepts and begin extending ideas about connection patterns as a prelude to the rest of the dissertation. The work presented here primarily uses unipartite networks (networks with only a single kind of vertex), but bipartite networks will also be discussed when necessary, so that it is not confusing when they come up in later chapters.



(a)

(b)

(c)

(d)

(e)

Figure 1.4: (a) Demonstrating a simple network. (b) A network with both directed edges (drawn with an arrow to indicate direction) and undirected edges. (c) A network exhibiting both multi-edges and weighted edges (drawn as thickness). (d) A network with multiple kinds of vertices. Connection patterns will be specified by the network – here the blue squares form a directed graph on their own, and the orange circles are only allowed to connect to blue squares and in an undirected fashion. (e) Bipartite network, a special case of the type of network shown in (d).

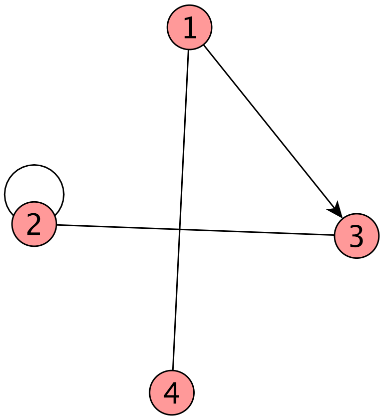
1.3.1 Adjacency matrix

The most common (and arguably most useful from a theoretical standpoint) mathematical way of describing a network, denoted G (for graph), is with the adjacency matrix [106]. For a network with n vertices, the adjacency matrix $\mathbf{A}$ is an $n \times n$ matrix where each row and column correspond to a particular vertex, and the particular elements correspond to the edges between the vertices. In the bipartite case, $\mathbf{A}$ is called the incidence matrix. The rows and columns correspond to the different kinds of vertices, so $\mathbf{A}$ is only square if there are equal numbers of the two kinds of vertices. $A_{ij} = 1$ means that there is an edge from the jth vertex to the ith vertex, and takes value 0 if no such edge exists. In the case of a weighted or multi-edge, A_{ij} takes value of the weight of the edge or number of edges respectively[3]. For undirected edges, $A_{ij} = A_{ji}$, and if the network is directed and it is the case that $A_{ij} = A_{ji}$, the edge is called reciprocated. By convention, an undirected self-edge takes the value 2 rather than 1, so that the number of ends of edges in the network is preserved when counting the elements of the adjacency matrix. An example of a network and its corresponding adjacency matrix is shown in figure 1.5a and b respectively.

It is common to have a bipartite network yet only be interested in one of the vertex types. In this case, most scientists project the network into a unimodal form, declaring an edge between two vertices if both connect to any of the same vertices in the bipartite case. In terms of the incidence matrix, this means taking the new adjacency matrix to be $\mathbf{A}\mathbf{A}^{\mathrm{T}}$ or $\mathbf{A}^{\mathrm{T}}\mathbf{A}$, where $^{\mathrm{T}}$ is the transpose operator, depending on which set of vertices is desired in the projection. Additionally, the projected network will often be forced to be simple by dropping all of the self-edges and setting all the remaining positive elements of the adjacency matrix to 1 to eliminate multi-edges.

With the adjacency matrix in hand, other important quantities of the network

[3] Care must be taken here! While weighted and multi-edges are significantly different conceptually and mathematically, they are represented the same way in the adjacency matrix.



(a)

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 2 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

(b)

Figure 1.5: A network (a) and its corresponding adjacency matrix (b) demonstrating both directed and undirected edges as well as a self-edge.

can be defined. The degree k_i of a vertex is the number of ends of edges connected to vertex i. In terms of the adjacency matrix, $k_i = \sum_j A_{ij}$. Furthermore, since each edge has 2 ends, the degrees relate to the number of edges m in the network by $\sum_i k_i = 2m$. In directed networks, it is sometimes important to distinguish the in- and out-degrees: $k_i^{\text{in}} = \sum_j A_{ij}$ and $k_i^{\text{out}} = \sum_j A_{ji}$. In this case, $\sum_i k_i^{\text{in}} = \sum_i k_i^{\text{out}} = m$ since the start and end of each edge are distinguishable.

The importance of the degree sequence, defined by the set $\{k_i\}$, cannot be understated. A large amount of information in the network is contained in this set. A related yet different concept is the degree distribution, studied by Rapoport and Horvath, which gives the probability that a vertex chosen uniformly at random from the network has a certain degree [124]. In many real-world networks, the degree distribution follows a long-tailed distribution[4] [102, 106]. In this case, it is common to

[4] One naturally wants to call this a power-law distribution. This dissertation will refrain from

have a small fraction of the vertices connecting with a large fraction of the edges, demonstrating the importance of the concept of degree in the network.

By definition of m and n, the average degree in the network $\langle k \rangle = \frac{2m}{n}$. The edge density is the fraction of possible edge pairs that are actually edges, i.e. the probability that a randomly chosen vertex pair have an edge: $\frac{2m}{n(n-1)}$. A network is said to be dense if $\langle k \rangle = \mathcal{O}(n)$ so that the edge density is a non-vanishing fraction with n. In all other cases, the network is called sparse. It is common to deal with an even more stringent limit, where $\langle k \rangle = \mathcal{O}(1)$. Unless otherwise stated, the latter behavior is assumed. When discussing real networks, the network is said to be sparse if $\langle k \rangle$ is only a small fraction of the possible n, since it does not make sense to talk about how the degree is changing with the size of the network given that it is a set size.

1.3.2 Transitivity and clustering

A precursor concept to communities (soon to be described in detail) is the clique. A clique in a network is a group of vertices in which every pair of vertices is connected by an edge (self-edges excluded). In terms of the adjacency matrix, a set C is a clique if for all $i, j \in C$ where $i \neq j$, $A_{ij} \neq 0$. A clique is considered to be maximal if no other vertices can be added to the group and still preserve the group's status as a clique. A different yet relevant concept regarding local graph structure is the neighborhood of a vertex, the set of vertices the vertex is connected to by edges. Many methods for detecting communities are based on these two concepts [9, 112].

Cliques with 3 vertices in them, generally referred to as triangles due to the way they are drawn in a network, hold a special interest for network scientists. A triangle is a potential indicator of transitivity in the network, the concept that the friend of

using this substitution however, as it has been proven that distinguishing a power law distribution from a log-normal distribution can be a difficult task [30]. Deviations at the upper end of the distribution are also observed with regularity so that neither of these distributions are an accurate description over the full range of degrees.

my friend is also (or should be) my friend. The amount of transitivity in a network is measured by computing the clustering coefficient, C [143, 146]. C is defined as the probability that a randomly selected connected triple (three vertices connected by two edges) is actually a closed triangle, and can be computed either on a vertex by vertex basis or over the entire network. For sparse networks, the density of edges is $\langle k \rangle / n$, which is small, so it would be reasonable to expect that the clustering should also be small. However, since the 1940s, it has been repeatedly noted that the clustering coefficient takes surprisingly high values in many types of networks, with values as high as 0.3 not uncommon [106, 107, 123]. One idea to explain transitivity in networks is triadic closure, the idea that a triangle became closed because two of the three relevant vertex pairs were already connected [34, 74, 99].

Triangles can also be discussed in a directed graph, but it is much more common to talk about motifs, sometimes referred to as triples in the sociology literature [72, 95, 134]. Historically, sociologists would compare sociograms by computing each of the sixteen possible unique configurations of directed edges, called motifs, between three vertices on the network. These calculations were spurred by a lack of computational power, while still being relatively informative about the network structure[5].

1.3.3 Paths and components

The extension of the concept of neighborhoods in a network is the path. A path is said to exist from vertex i to vertex j if it is possible follow edges in the network starting at i and end at j [106]. In terms of the adjacency matrix, a path of length d from i to j is one where there exists some set $\{l_1, l_2, \ldots, l_{d-1}\}$ such that $A_{jl_1} A_{l_1 l_2} \cdots A_{l_{d-2} l_{d-1}} A_{l_{d-1} i} \neq 0$. A path from i to itself is called a cycle. While this last concept is essentially useless in an undirected network without adding conditions about backtracking along edges, it is always non-trivial in directed networks. A

[5] Ironically, this is a computationally expensive calculation to do, taking $\mathcal{O}(n^3)$ time to complete.

directed network in which there are no cycles is called a directed acyclic graph.

The distance between two vertices is the length of the shortest path between them, which itself is referred to as a geodesic[6]. The average path length is the average distance between vertices in the network. The diameter of the network is the maximum distance taken over all pairs of vertices in the network. The diameter is difficult to compute on large networks, so the average distance is often used as a proxy even though the two can be quite different.

The concept of paths in networks has made it into popular culture. In the late 1960s, a psychologist by the name of Stanley Milgram ran a famous experiment where he asked about 300 residents of Kansas and Nebraska to try to reach a particular friend of his in Boston, with the condition that they were only allowed to forward the letter and directions to a direct acquaintance, who would then continue the chain [94]. Of the forty or so letters that completed the journey, on average they had been forwarded 6.2 times, motivating the popular phrase "six degrees of separation" [145]. Although the methodology in his experiment was questionable (What happened to the letters that did not make it? For the ones that did, did they take the shortest path? What about selection bias in the sample?), many similar results quickly started appearing with other networks [24]. This phenomenon has since been termed the small world effect, which notes that the average path length in a network generally grows slowly with the size of the network [146].

A connected component in a network is the maximal set of vertices where any two vertices have a finite length path between them (in fact, in an undirected network, a vertex that connects to any vertex in the connected component must itself be in the connected component) [106]. Generally there is at most a single connected component that takes up a significant fraction of the vertices in the network, called the giant component, and a number of small components that take varying sizes but

[6] Note that there may be more than one geodesic between two vertices.

all scale independently of n [102]. Calculations in this dissertation will typically be restricted to the giant component since it is the most interesting part of the network for the problems discussed.

In a directed network, there are four kinds of partially nested component types. The weakly connected component of a vertex consists of the connected component ignoring edge direction. This contains both the in- and out-components, which are the set of vertices that can reach/be reached by the vertex in question respectively[7]. The strongly connected component (SCC) is the closest analog of the connected component from the undirected case, which is the set of vertices that can both reach and be reached by other vertices in the SCC – the intersection of the in- and out-components. In other words, for any two vertices in the SCC, there must exist a cycle for one of the vertices that contains the other vertex. As with undirected networks, it is commonly found that there is at most one giant component and many small components, each represented in the form of a bow tie diagram that demonstrates the different component types [22].

1.3.4 Community detection

Continuing the trend of expanding the scale of focus of connection patterns, the scale within a network called the intermediate scale, referred to in Physics as the mesoscopic scale, will be discussed. In real networks, connections between vertices are not random. Vertices have internal characteristics that play into how they connect with their neighbors. A common dynamic in many networks is assortative mixing, sometimes called homophily: actors in the network with a particular property are more likely to connect to others with the same property. For example, students are more likely to be friends with other students of the same age, sex, or race [97, 124].

[7] However, it is incorrect to think of the weakly connected component as the union of the in- and out-components, as this does not necessarily include all of the vertices in the weakly connected component.

When studying networks, the internal mechanisms that generate the network are usually not known, but the effect is seen. Intuitively, the converse of the previous paragraph should be true: the connection patterns should be informative about the vertices in the network, and this is the goal of community detection. For a specific subset of vertices, it is said to be a community if the vertices primarily connect to other vertices within the subset [54, 58]. An example of such a structure is shown in Figure 1.6. While researchers all agree that this structure must be true in a loose sense, the precise definition of a community is subject to much debate [54]. Depending on the context, communities can even be disjoint or overlapping. Community structure and the associated network mechanisms is an important topic for network scientists and a lot of work has gone into studying the subject. For the reader interested in the subject who wants significant detail on the problem and the work done to date, Fortunato gives an excellent review of the entire subject in [54].

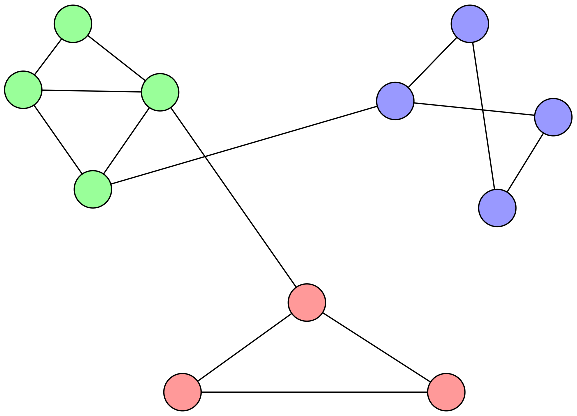


Figure 1.6: Example of a network with communities. The vertices are colored according to which of the three separate communities they belong.

There are several early methods for community detection worth mentioning. The first is hierarchical clustering, which arose out of sociological interests. The idea behind hierarchical clustering is that similar vertices (for however similar is defined in the particular context) should slowly aggregate in a pairwise fashion until eventually the entire network has aggregated into one single group. Then the objective is to find the right scale at which to stop the agglomeration. Similarly hierarchical clustering can be run by dividing up the network until each vertex is in its own group, which is known as a divisive algorithm. Whether agglomerative or divisive, the full algorithm is represented as a picture in the form of a dendrogram, a form of tree where all the vertices are lined up in a row and connections up the tree show the different agglomeration/division steps. The final cut is a horizontal line where the connections above the line are ignored and the connections below the line form the communities. An example of this technique is the method by Newman and Girvan [108].

The second method is graph partitioning, a method which originated in computer science. The goal of graph partitioning is to minimize the number of edges that run between two groups of a given size, i.e. to find the groups of vertices that require the fewest number of removed edges to create two separate connected components. This is commonly done by computing the Laplacian matrix, a positive semi-definite matrix closely related to the adjacency matrix [29]. The zero eigenvalues of the Laplacian matrix have associated eigenvectors that correspond to the different connected components of the network. Thus intuitively, two groups of vertices that only have a few edges between them should have a fairly small corresponding eigenvalue, and many graph partitioning methods are based around finding these small eigenvalues [52].

One of the more famous recent measures for community detection is modularity, which grew out of the Girvan-Newman algorithm mentioned above [108]. The idea behind modularity is that two vertices connected by an edge in the same community should get a positive score, but receive a penalty if the two vertices are not connected by an edge, so that a good set of communities gives a high modularity score. The exact amount of penalty can be changed depending on the preferred community structure, but is most commonly taken to be the expected number of edges between the two vertices under the configuration model (to be described in section 1.3.5) so that the trivial grouping of the entire network being one community gives a modularity of 0. In this case the expression for modularity can be written as $Q = (A_{ij} - \frac{k_i k_j}{2m})\delta(g_i, g_j)$, where g_i is the community of vertex i and δ is the Kronecker delta[8].

In an ideal world, it would be possible to find the best modularity score over all combinations of communities, but this has been proven to be difficult [21]. Many computational methods have been devised to approximately maximize modularity, and in practice this measure and the numerous methods do a very good job of finding community structure in networks [54]. Modularity does have a few known problems however, for instance it cannot be used to find very small communities [55], may not have a unique optimum [59], and is somewhat unsatisfactory from a formal viewpoint [17, 155].

Sometimes it is desired to discuss variants of community structure. As mentioned earlier in the section, there is a problem known as overlapping community detection where vertices are allowed to belong to multiple groups. An early method created for detecting overlapping communities is clique percolation [112]. This method looks for cliques with c vertices in them (where c is an input to the method), and defines two such cliques to be adjacent if they share $c - 1$ vertices. Then a community is the set of vertices that can be reached by traversing adjacent cliques with c vertices as though they were paths. Since a vertex can be in multiple cliques with c vertices that are not necessarily adjacent, this method inherently gives overlapping communities.

Many methods like clique percolation are based on the local structure of the network [9]. These methods are based around growing communities in the network rather than splitting the network up into communities. This has the advantage of finding communities that are compact, connected, and potentially overlapping, several aspects of which are not guaranteed by global methods. The number of communities

[8] Modularity is often written with a normalizing factor of $\frac{1}{2m}$ so that it is restricted to run between -1 and 1, although this does not affect the extracted community structure for a given network in any way since it is just a multiplicative constant.

that local methods find is also dependent on the graph structure rather than being a set number, which could be seen as an advantage or disadvantage depending on the application[9]. It may also be desirable to allow for disassortative structure in the community (for example as found in a bipartite network), something which these local methods cannot address. Many of these points will be discussed in further detail throughout this dissertation.

1.3.5 Generative Models

A useful technique in the study of complex networks are generative models, models that can be used to generate synthetic networks [71, 73, 109]. The purpose of using these models is to generate example networks that have particular desirable properties [13, 96, 105, 117, 146]. These can then be used as benchmarks for testing algorithms against or to compare synthetic and real networks.

Many generative models assume that all edges are placed independently[10]. While realistically this may seem a very poor assumption to make since, as noted earlier in the chapter, there is significant clustering in real networks, it is still possible to get useful information out of the models[11]. Furthermore, this makes the math involved more tractable. Throughout the body of this dissertation, this will always be assumed. It is also common to assume the limit of large network size with respect to the vertices, although this assumption will not be the case outside of this section unless stated otherwise.

Historically, one of the simplest, earliest random network model is the Erdős-Rényi random graph, which comes in two flavors [47, 48, 49, 133]. In the first version of

[9] Ideally there should be method that can handle either a fixed or an unknown number of communities, but little work has been done on this difficult problem [121].

[10] There are a few especially notable exceptions to this. For example, the configuration model [16, 96, 111] has small correlations between edges. There is also a class of models designed specifically to create networks with non-vanishing clustering. These models place triangles directly into the network instead of edges [105].

[11] This is possibly the most surprising and interesting result to come out of modern statistics, that unrealistic assumptions can still give useful results.

this model, $G(n,p)$, an edge is placed between each pair of vertices with probability p, where p is the same for all vertex pairs. To borrow terminology from physics, this model is a canonical ensemble on the edges, and the second version of the Erdős-Rényi random graph, $G(n,m)$, is the microcanonical ensemble equivalent. $G(n,m)$ generates m edges and places them uniformly at random over the unique vertex pairs without replacement. In the limit of large network size, these two models are equivalent under the substitution $n^2p = 2m$.

Since edges are placed independently, it is trivial to see that in the Erdős-Rényi model, the expected degree for each vertex is merely $\langle k_i \rangle = \langle k \rangle = np$. As mentioned in section 1.3.1, the degree sequence is very informative, so this model is not especially realistic[12].

A more realistic model is the well-known configuration model, invented by Malloy and Reed, Bender and Canfield, and many other mathematicians [16, 96, 111]. In this model, the vertices have a given number of edge stubs so as to preserve the degree, and the edges are placed uniformly at random without replacement over the stubs. Thus the probability of an edge between two vertices i and j is $\frac{k_i k_j}{2m}$ where k_i are the parameters of the model and $2m = \sum_i k_i$ by definition, although this is a misnomer since the graph is not actually generated in this fashion. A modified version of the configuration model with this interpretation in mind was introduced by Chung and Lu [28]. In this version of the model, edges are placed independently between two vertices following a Poisson distribution with mean equal to $\frac{k_i k_j}{2m}$ so that the degree is preserved only in expectation. In this regard, the degree sequence in this model is analogous to a canonical ensemble from statistical physics compared to the configuration model's microcanonical ensemble. The advantage of the Chung-Lu model over the configuration model is that it is occasionally easier to deal with it mathematically.

[12] In fact this model is not realistic in a variety of ways. About the only property it does correctly match to real networks (besides the total number of edges) is the small-world effect.

For most probabilistic network models that place edges independently, each edge follows a Bernoulli distribution with probability dictated by the model; simply put, either there is an edge between a pair of vertices or there is not an edge. In some cases, it is more convenient to use the Poisson distribution where the parameter of the distribution, which doubles as the mean, is the same value as the Bernoulli case, where it also doubles as the mean. Technically this creates a multigraph, but for sparse networks the expected number of multi-edges is small, so it is only a minor error. In return, the mathematics can become significantly easier due to the additive property of the Poisson distribution: the sum of two Poisson distributed random variables with means λ_1 and λ_2 respectively is Poisson distributed with mean $\lambda_1 + \lambda_2$. This dissertation will repeatedly take advantage of this formulation.

A common way of generating networks, rather than placing edges probabilistically, is to grow them: the network starts with a small number of vertices, and then more are added and edges are placed between the vertices that exist at each step. A common theory is that some networks follow a rich get richer scheme as they develop, where vertices with a higher degree are more likely to have new edges connect to them [120]. This is commonly thought to be true for citation networks, where it is only possible to cite papers that have already been written and scientists are more likely to cite the famous papers than lesser-known ones in their field. This preferential attachment has been realized in many network models, for example in the famous paper by Barabási and Albert [13, 117]. In this model, at each time step, a single vertex gets added to the network and has a set degree. The other ends of the new edges get placed among the existing vertices in proportion to their degree. In particular, this model reproduces a long-tailed degree distribution, a feature observed in real networks [120].

Another common synthetic modeling technique is to start with a network that has a regular structure and move the edges around, called rewiring. One well-known model of this type is the small-world model by Watts and Strogatz [146]. In this model, the n vertices each have degree $2k$, where the vertices are arranged in a circle and connect to k neighbors on each side. Then with probability p, each edge is removed and placed again uniformly at random over the vertex pairs. Depending on the value of p (and k, albeit in a minor role), the resulting network can have zero or non-zero clustering and exhibits or does not exhibit the small world effect.

With these and many more models, it is possible to calculate all sorts of interesting statistics. However, this dissertation will take a different turn; the main focus is in fitting network models to real networks. Network models are perfect for learning the structure of real networks since they are well-principled – the type of structure the model is looking for can be understood exactly. In particular, since the models presented in this dissertation are probabilistic in nature, the statistics literature can be leveraged against all of these questions. Likelihood maximization methods are a perfect place to start this discussion.

1.3.6 Maximum likelihood estimation

A commonly used method for extracting information from networks by using probabilistic models is maximum likelihood estimation. The idea behind this method is rather simple: maximize the probability (referred to in this sort of posterior case as the likelihood) that the observed graph was generated with respect to the parameters of the model. According to most models, the edges are generated independently, so this will often take the form

$$P(G|\boldsymbol{\Theta}) = \prod_{i<j} P(A_{ij}|\boldsymbol{\Theta}) \prod_{i} P(A_{ii}|\boldsymbol{\Theta}) \tag{1.1}$$

for an undirected network where $\boldsymbol{\Theta}$ are the model parameters. The self-edge terms, when allowed as in equation 1.1, are split into a separate term and will look slightly different compared to the other edge terms because of the conventional factors of two in the adjacency matrix.

Since the likelihood is a product of a large number of terms, it is more convenient to work with its logarithm, called the log-likelihood, which turns the product into a sum without changing the position of the maximum because the logarithm function is strictly increasing. For simple models, the maximization can be done directly by taking derivatives, setting the expression equal to 0, and solving it in the standard Calculus-based approach. For example, the Erdős-Rényi random graph $G(n, p)$ (ignoring self-edges for simplicity) has likelihood

$$P(G|\mathbf{\Theta}) = \prod_{i<j}\Big(p^{A_{ij}}(1-p)^{(1-A_{ij})}\Big) \tag{1.2}$$

where the product only runs over $i < j$ since the model generates undirected graphs. Taking the log of equation 1.2 gives us the log-likelihood

$$\mathcal{L} = \sum_{i<j}\Big(A_{ij}\log(p) + (1-A_{ij})\log(1-p)\Big). \tag{1.3}$$

Setting the derivative of the log-likelihood with respect to p equal to 0 and solving for p gives the best fit solution:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial p} = 0 &= \sum_{i<j}\left(\frac{A_{ij}}{p} - \frac{(1-A_{ij})}{1-p}\right)\\ 0 &= \frac{m}{p} - \frac{\binom{n}{2} - m}{1-p}\\ p &= m/\binom{n}{2}. \qquad (1.4)\end{aligned}$$

Fitting this model to an observed network matches the number of edges to the number observed in the graph (or equivalently matches the average degree) but cannot do anything more complex than that.

A maximum likelihood approach is used in many situations where a probabilistic model is fit to data. However, direct maximization alone will not be sufficient for the models presented in this dissertation since, as will be seen in the next section, many of the parameters are discrete, so other methods must be employed to estimate those particular parameters.

1.3.7 Stochastic blockmodels

The natural extension of the previous sections is the family of models known as stochastic blockmodels [131, 142]. In these models, each vertex is assigned to a group, and connect with the other vertices based on these groups. As with other probabilistic network models, the strength of the stochastic blockmodels stems from their rigorous mathematical background. It is possible to prove exactly what kind of structure, community or otherwise, the model is picking out, and when that structure can be found [37, 38]. They also offer a flexibility unrivaled by other methods – since much of the information is encoded by the vertices themselves, they can be used to describe a wide variety of network structures with minimal modification.

An early and particularly simple model in this class is the standard stochastic blockmodel, a community detection model where the probability of an edge between two vertices is given by a mixing matrix, ω_{g_i,g_j} whose element is determined by the communities of the two vertices, denoted g_i [131, 142]. The name of this model and the class of models in general is a reference to the sociological model mentioned in section 1.1, although it is clear that this model is significantly more flexible than the earlier one. For example, it can account for both assortative and disassortative structure depending on the elements of the mixing matrix, whether they are strongly diagonal or strongly off-diagonal. However, like the Erdős-Rényi random graph, vertices within a group all have the same expected degree. What this means in practice is that if this model is used to find community structure in a real network, the best set of "communities" it finds often correspond to a split between high and low-degree vertices [37].

A significantly more useful model is Karrer and Newman's degree-corrected stochastic blockmodel [83]. In this model, every vertex has an additional parameter θ_i that controls the degree of the vertex, while retaining the mixing matrix ω from the earlier model. The probability of an edge between two vertices is $\theta_i \omega_{g_i,g_j} \theta_j$. The additional flexibility in the form of controlling the degree makes this model much more useful in practice than the standard stochastic blockmodel.

The models discussed in the body of this dissertation are all variants of stochastic blockmodels. The discussion in this section has focused on traditional assortative and disassortative community detection up to this point, but it should be clear that this sort of model can be useful in discovering other structure as well. Such an example will be discussed in depth in chapter V in the form of a mathematical realization of Homans' ranking model [75].

1.3.8 Expectation-maximization algorithm

As mentioned in section 1.3.6, maximization via derivatives is not sufficient for many network models. To supplement that approach, this dissertation will make extensive use of the maximization technique known as the expectation-maximization (EM) algorithm [39]. The EM algorithm is a technique for maximizing the likelihood of a parameterized latent, that is unobserved, variable model. For stochastic blockmodels, the latent variables are the communities (or whatever vertex information the model concerns itself with) and the parameters are the mixing matrix and any other relevant parameters explicitly defined in the model. The algorithm is performed by splitting the single maximization into two deterministic steps, which individually can be much simpler to solve than the combined problem. The first (Expectation) step is to find the distribution of the latent variables while holding all other observables and parameters constant, and the second (Maximization) step is to maximize the explicit expression with the latent variables over the parameters, a step which is often done directly with derivatives. These two steps are computed in an alternating fashion until the parameters converge, at which point both steps are satisfied simultaneously. The EM algorithm is proven to monotonically increase the likelihood at each step, and converge to a critical point of the likelihood[13]. This critical point is not guaranteed to be the absolute maximum, so the algorithm will typically be run multiple times (at different starting locations, since it is deterministic) and the best result kept as the desired answer.

There are many equivalent formulations of the EM algorithm, and two will be presented here which will be useful later in the dissertation. Mathematically the objective is to turn the log of a sum into a sum of logs, since these are much easier to differentiate. A simple and direct way this can appear is via Jensen's inequality in the form

$$\log\big(\textstyle\sum_u x_u\big) \geq \sum_u q_u \log \frac{x_u}{q_u}, \tag{1.5}$$

where the x_u are some set of positive numbers (which in our case will be related to the probability of a specific edge) and the q_u are any nonnegative numbers satisfying $\sum_u q_u = 1$. This statement is a valid application of Jensen's inequality because the logarithm function is concave. Notice in particular that the equality can be recovered by making the particular choice

$$q_u = x_u / \sum_u x_u. \tag{1.6}$$

It is not immediately clear how the q_u correspond to the latent variables of our models, so this is demonstrated by comparing this method with a second way of writing the

[13] Since each step individually maximizes the log-likelihood with respect to a particular set of parameters, it is trivial that the likelihood must increase with each iteration. Furthermore, well-defined likelihoods have an upper bound of 1, so this increase has to stop somewhere. It will soon be shown that where this two-step process converges corresponds to a critical point of the original likelihood.

EM algorithm:

$$\log(P(G|\boldsymbol{\Theta})) \geq \log(P(G|\boldsymbol{\Theta})) - \mathrm{D}\big(Q(Z)||P(Z|G,\boldsymbol{\Theta})\big), \tag{1.7}$$

where $P(G|\boldsymbol{\Theta})$ is the probability of the data given the parameters (written in a format suggestive to the application towards networks), Z refers to the latent variables, and we've introduced $Q(Z)$ as any probability distribution over Z. $\mathrm{D}(P_1(X)||P_2(X)) = \sum_X P_1(X)\log\Big(\frac{P_1(X)}{P_2(X)}\Big)$ is the Kullback-Leibler divergence (also referred to as the relative entropy) of distributions $P_1(X)$ and $P_2(X)$ over some random variable X. In particular, the Kullback-Leibler divergence is weakly greater than 0 and equals 0 if and only if $P_1(X) = P_2(X)$ almost everywhere[14]. Thus whereas $Q(Z)$ can be any probability distribution, the right hand side of equation (1.7) will be maximized (and thus the two sides of the equation will be equal) when $Q(Z) = P(Z|G,\boldsymbol{\Theta})$, the distribution of the latent variables. Simplifying equation (1.7) gives

$$\begin{aligned}\log(P(G|\boldsymbol{\Theta})) &\geq \log(P(G|\boldsymbol{\Theta})) - \sum_Z Q(Z)\log\Big(\frac{Q(Z)}{P(Z|G,\boldsymbol{\Theta})}\Big)\\ &= \sum_Z Q(Z)\Big[\log(P(G|\boldsymbol{\Theta})) - \log\Big(\frac{Q(Z)}{P(Z|G,\boldsymbol{\Theta})}\Big)\Big]\\ &= \sum_Z Q(Z)\Big[\log(P(G,Z|\boldsymbol{\Theta})) - \log(Q(Z))\Big]\\ &= \sum_Z Q(Z)\Big[\log\big(P(G|Z,\boldsymbol{\Theta})P(Z|\boldsymbol{\Theta})\big) - \log(Q(Z))\Big]. \end{aligned} \tag{1.8}$$

$P(G|Z,\boldsymbol{\Theta})$ is the form of our model given the latent variables and is much easier to write explicitly than $P(G|\boldsymbol{\Theta})$ since there's no additional sum over the latent variables. The term $P(Z|\boldsymbol{\Theta})$ is the prior on the latent variables and is often independent compared to the probability of the graph with respect to the parameters and thus maximized separately from the rest of the equation. The two steps of the EM algo-

[14] For the network models, the fact that the Kullback-Leibler divergence is additive over independent random multivariate variables will also be needed. This step is omitted here for simplicity.

rithm are to compute $Q(Z) = P(Z|G, \boldsymbol{\Theta})$ holding $\boldsymbol{\Theta}$ constant then to maximize equation (1.8) with respect to $\boldsymbol{\Theta}$ while holding $Q(Z)$ constant. Comparing equations (1.5) and (1.8) with the appropriate substitutions ($q_u = Q(Z)$ and $x_u = P(G, Z|\boldsymbol{\Theta})$), it is clear that the two are equivalent and thus both represent valid EM algorithms. This dissertation will primarily use the Jensen formulation since it will be easier to follow for the models presented.

In most applications of the EM algorithm, the M step becomes easy to compute and a closed form solution is not uncommon. The E step can be another story however. It is rare for this step to have a closed form solution, so a numerical method like Markov Chain Monte Carlo (MCMC) is usually needed to obtain the distribution of the latent variables [127].

1.4 Outline of the dissertation

In this chapter we have given a brief introduction to networks. The techniques discussed will be instrumental in the following chapters. With this dissertation, we expand upon previous work in several key areas of networks, focusing on the use of stochastic modeling techniques. Specifically, we apply techniques previously unseen to networks research and make progress on unanswered questions. We use principled approaches and when possible make rigorous derivations of our methods. The results are useful methods that are effective both in synthetic tests as well as real world networks.

In chapter II, we introduce a stochastic blockmodel for overlapping community detection that can be run in a memory and computationally efficient manner due to both steps of the EM algorithm having closed form solutions. Part of our effort is spent improving upon the efficiency of the implementation, making it useful for detecting community structure even on modern networks numbering in the millions of nodes and edges. We include appendices (A,B,C) tying this model with other models both in the networks and computer science literatures, and to demonstrate our implementation's efficiency. This chapter and associated appendices are based on the work presented in the author's publication [10].

In chapter III, we create a heuristic that attempts to solve the question of choosing the correct number of communities for the overlapping community detection model introduced in chapter II. This heuristic is based loosely on model selection techniques from Statistics, specifically likelihood ratio tests. Using networks generated according to this community detection model, we show that the heuristic correctly selects the number of communities where other model selection methods fail, and it can also accurately predict the mean of the change in log-likelihoods when comparing the fits for K and $K+1$ communities where no more than K communities are present in the network. Furthermore, we demonstrate that this accuracy in selecting the correct number of communities translates over to real networks. This chapter is based on unpublished work.

Chapter IV explores a different topic of interest, multi-modal networks. We analyze data from papers published in the American Physical Society journal series Physical Review, taking into account both authors and paper citations. We give special attention to how authors cite each others' papers, an analysis that requires both authorship and citation information. This is supplemented by publication date information, so we also study how these citation patterns have changed over the history of the journals. This temporal discrimination is especially important since the number of publications in the journals is growing exponentially. We discover that a researcher cites his or her own papers and his or her collaborators' papers a large fraction of the time, but the extent of this has not changed significantly over time. Furthermore, the citations received from collaborators come sooner than researchers not personally known. There is also a large amount of reciprocity; citations from one scientist to another are often returned in the form of citations later on. Transitivity among authors is large. However, while still significant, triadic closure is only a small part of transitivity on the whole. This chapter is based on the author's publication [92].

Chapter V focuses on a blockmodel which, instead of use in community detection, is used for rank ordering a network based on global flow of edges. The basic idea is that the type of network we are interested in is structured almost in an acyclic fashion, and we use our model to describe the generative processes involved. We apply this model on a collection of high school friendship networks and show that the networks demonstrate remarkably similar behavior, regardless of the characteristics of the school. The rank of the vertices appear to have a striking resemblance to a measure of popularity or social status within the school, and the pattern of connections qualitatively resembles Homans' tiered model [75]. This chapter is based on the author's publication [11].

CHAPTER II

Overlapping community detection model

2.1 Introduction

In chapter I, we introduced background on community detection methods. We gave two examples of stochastic blockmodels that are used to detect communities in the traditional sense. In this chapter, we attempt to extend these methods to allow for overlapping communities. In a general sense, overlapping communities are significantly more difficult to describe than nonoverlapping communities because of a much larger number of parameters – the models can describe a wider variety of structure. Furthermore, even if you're able to write down a model, it is possible that you won't be able to extract any useful information in a reasonable amount of time.

One way researchers have worked around this problem is to create methods based on local community structure [9]. Rather than splitting an entire network into communities in one step, these methods instead look for local groups within the network, based on analysis of local connection patterns. Methods of this kind give rise naturally to overlapping communities when one generates a large number of independent local communities throughout the network. Moreover, the communities tend to be compact and connected subgraphs, a requirement not always met by other methods. On the other hand, global detection methods can capture large-scale network structure better and are more appropriate when particular constraints, such as constraints on the number of communities, must be satisfied. Furthermore, local methods do not have the nice property of blockmodels that they can be used and studied from a generative standpoint.

There have been some advances for using network models for overlapping community detection, but most previous work on this subject use "mixed membership" models [4], in which, typically, vertices can belong to multiple groups and two vertices are more likely to be connected if they have more than one group in common. This, however, implies that the area of overlap between two communities should have a higher average density of edges than an area that falls in just a single community. It is unclear whether this reflects the behavior of real-world networks accurately, but it is certainly possible to construct networks that do not have this type of structure. Ideally we would prefer a less restrictive model that makes fewer assumptions about the structure of community overlaps.

In this chapter, we develop a global statistical method for detecting overlapping communities based on the idea of link communities, which has been proposed independently by a number of authors both in the physics literature [2, 50] and in machine learning [64, 115]. The idea is that communities arise when there are different types of edges in a network. In a social network, for instance, there are links representing family ties, friendship, professional relationships, and so forth. If we can identify the types of the edges, i.e., if we can group not the vertices in a network but the edges, then we can deduce the communities of vertices after the fact from the types of edges connected to them. This approach has the nice feature of matching our intuitive idea of the origin and nature of community structure while giving rise to overlapping communities in a natural way: a vertex belongs to more than one community if it has more than one type of edge. Such an approach also requires a shift in thought on the generative side of things: whereas the standard vertex communities can be defined before the edges are placed, link communities must generate the edges and communities simultaneously.

We define a model in this chapter with these criteria in mind. This model has an EM algorithm where both steps are closed-form, giving each iteration a linear computational complexity, and we discuss how the algorithm can be implemented to optimize speed and memory requirements, allowing applications to large networks. We give example applications to numerous real-world networks, as well as tests against synthetic networks that demonstrate that the algorithm can discover known overlapping community structure in such networks.

We also show how our method can be used also to detect nonoverlapping communities by assigning each vertex solely to the community to which it most strongly belongs in the overlapping division. We demonstrate that this intuitive heuristic can be justified rigorously by regarding the link community model as a relaxation of the degree-corrected stochastic blockmodel [83]. Algorithms have been proposed previously for fitting this blockmodel, but their running time was typically at least quadratic in the number of vertices, which limited their application to smaller networks. The algorithm proposed here is significantly faster and hence can be applied to the detection of disjoint communities in very large networks.

2.2 A generative model for link communities

Our first step is to define the generative network model that we will use. The model generates networks with a given number n of vertices and undirected edges divided among a given number K of communities. It is convenient to think of the edges as being colored with K different colors to represent the communities to which they belong. Then the model is parameterized by a set of parameters θ_{iu}, which represent the propensity of vertex i to have edges of color u. Specifically, $\theta_{iu}\theta_{ju}$ is the expected number of edges of color u that lie between vertices i and j, the exact number being Poisson distributed about this mean value. Note that this means the network is technically a multigraph—it can have more than one edge between a pair of vertices. Some real-world networks contain such multiedges: in network representations of the world wide web, for instance, a single web page can contain several hyperlinks to the same other page. Most networks, however, have single edges only, and in this sense the model is unrealistic. However, allowing multiedges makes the model enormously simpler to treat and in practice the number of multiedges tends to be small, so the error introduced is also small, typically vanishing as $1/n$ in the limit of large network size. Multiedges are also allowed in most other random graph models of networks, such as the widely studied configuration model [96, 111], and are neglected there for the same reasons. Our model also allows self-edges—edges that connect to the same vertex at both ends—with expected number $\frac{1}{2}\theta_{iu}\theta_{iu}$, the extra factor of a half being convenient for consistency with later results. Again, the appearance of self-edges, while unrealistic in some cases, greatly simplifies the mathematical developments and introduces only a small error.

In the model defined here the link communities arise implicitly as the network is generated, as discussed in the introduction, rather than being spelled out explicitly. Two vertices i, j which have large values of θ_{iu} and θ_{ju} for some value of u have a high probability of being connected by an edge of color u, and hence groups of such vertices will tend to be connected by relatively dense webs of color-u edges—precisely the structure we expect to see in a network with link communities.

2.3 Detecting overlapping communities

Given the model defined above, it is now straightforward to write down the probability with which any particular network is generated. Recalling that a sum of independent Poisson-distributed random variables is also a Poisson-distributed random variable, the expected total number of edges of all colors between two vertices i and j is simply $\sum_u \theta_{iu}\theta_{ju}$ (or $\frac{1}{2}\sum_u \theta_{iu}\theta_{iu}$ for self-edges), and the actual number is

Poisson-distributed with this mean[1]. Thus the probability of generating a graph G with adjacency matrix elements A_{ij} given the set of parameters $\boldsymbol{\Theta}$ is

$$
\begin{aligned}
P(G|\boldsymbol{\Theta}) = {} & \prod_{i<j} \frac{\bigl(\sum_u \theta_{iu}\theta_{ju}\bigr)^{A_{ij}}}{A_{ij}!} \exp\Bigl(-\sum_u \theta_{iu}\theta_{ju}\Bigr) \\
& \times \prod_i \frac{\bigl(\frac{1}{2}\sum_u \theta_{iu}\theta_{iu}\bigr)^{A_{ii}/2}}{(A_{ii}/2)!} \exp\Bigl(-\tfrac{1}{2}\sum_u \theta_{iu}\theta_{iu}\Bigr).
\end{aligned}
\tag{2.1}
$$

(Recall that the adjacency matrix element A_{ij}, by convention, takes the value $A_{ij} = 1$ if there is an edge between distinct vertices i and j, but $A_{ii} = 2$ for a self-edge—hence the additional factors of $\frac{1}{2}$ in the second product.)

We fit the model to an observed network by maximizing this probability with respect to the parameters θ_{iu}, or equivalently (and more conveniently) maximizing its logarithm. Taking the log of Eq. (2.1), rearranging, and dropping additive and multiplicative constants (which have no effect on the position of the maximum), we derive the log-likelihood

$$
\log P(G|\boldsymbol{\Theta}) = \sum_{ij} A_{ij} \log\bigl(\textstyle\sum_u \theta_{iu}\theta_{ju}\bigr) - \displaystyle\sum_{iju} \theta_{iu}\theta_{ju}. \tag{2.2}
$$

Direct maximization of this expression by differentiating leads to a set of nonlinear implicit equations for θ_{iu} that are hard to solve, even numerically. An easier approach is the following. We apply Jensen's inequality in the form[2]:

$$
\log\bigl(\textstyle\sum_u x_u\bigr) \geq \displaystyle\sum_u q_u \log \frac{x_u}{q_u}, \tag{2.3}
$$

[1] Another way of viewing this model is that each vertex i has a community membership vector $\theta_{\mathbf{i}}$, and the expected number of edges between two vertices is the inner product of their community vectors. This interpretation illuminates a rotational symmetry for the parameters in the model, so for consistency with later interpretations, the rotation that puts all of the vertices' community vectors in the first sectant will be taken. See Appendix D for why we can always make this choice.

[2] This is a special case of the general observation that the log of the average of any set of numbers is never less than the average of the log, since the logarithm function is concave down. If the numbers in question are x_u/q_u and the average is taken with weights q_u this observation leads immediately to the inequality given.

where the x_u are any set of positive numbers and the q_u are any probabilities satisfying $\sum_u q_u = 1$. Note that the exact equality can always be achieved by making the particular choice $q_u = x_u / \sum_u x_u$. Applying Eq. (2.3) to Eq. (2.2) gives

$$\log P(G|\mathbf{\Theta}) \geq \sum_{iju} \left[A_{ij} q_{ij}(u) \log \frac{\theta_{iu}\theta_{ju}}{q_{ij}(u)} - \theta_{iu}\theta_{ju} \right], \tag{2.4}$$

where the probabilities $q_{ij}(u)$ can be chosen in any way we please provided they satisfy $\sum_u q_{ij}(u) = 1$. Notice that the $q_{ij}(u)$ are only defined for vertex pairs i, j that are actually connected by an edge in the network (so that $A_{ij} = 1$), and hence there are only as many of them as there are observed edges.

Since, as noted, the exact equality in this expression can always be achieved by a suitable choice of $q_{ij}(u)$, it follows that the double maximization of the right-hand side of (2.4) with respect to both the $q_{ij}(u)$ and the θ_{iu} is equivalent to maximizing the original log-likelihood, Eq. (2.2), with respect to the θ_{iu} alone. It may appear that this does not make our optimization problem any simpler: we have succeeded only in turning a single optimization into a double one, which one might well imagine was a more difficult problem. Delightfully, however, it is not; the double optimization is actually very simple. Given the true optimal values of θ_{iu}, the optimal values of $q_{ij}(u)$ are given by

$$q_{ij}(u) = \frac{\theta_{iu}\theta_{ju}}{\sum_u \theta_{iu}\theta_{ju}}, \tag{2.5}$$

since these are the values that make our inequality an exact equality. But given the optimal values of the $q_{ij}(u)$, the optimal θ_{iu} can be found by differentiating (2.4), which gives

$$\theta_{iu} = \frac{\sum_j A_{ij} q_{ij}(u)}{\sum_i \theta_{iu}}. \tag{2.6}$$

Summing this expression over i and rearranging gives us

$$\Big(\sum_i \theta_{iu}\Big)^2 = \sum_{ij} A_{ij} q_{ij}(u), \tag{2.7}$$

and combining with (2.6) again then gives

$$\theta_{iu} = \frac{\sum_j A_{ij} q_{ij}(u)}{\sqrt{\sum_{ij} A_{ij} q_{ij}(u)}}. \tag{2.8}$$

Maximizing the log-likelihood is now simply a matter of simultaneously solving Eqs. (2.5) and (2.8), which can be done iteratively by choosing a random set of initial values and alternating back and forth between the two equations. This type of approach is known as an expectation-maximization or EM algorithm and it can be proved that the log-likelihood increases monotonically under the iteration, though it does not necessarily converge to the global maximum. To guard against the possibility of getting stuck in a local maximum, we repeat the entire calculation a number of times with random initial conditions and choose the result that gives the highest final log-likelihood. In the work presented here we found good results with numbers of repetitions in the range from 10 to 100.

The value of $q_{ij}(u)$ in Eq. (2.5) has a simple physical interpretation: it is the probability that an edge between i and j has color u, which is precisely the quantity we need in order to infer link communities in the network. Notice that $q_{ij}(u)$ is symmetric in i, j, as it should be for an undirected network.

The calculation presented here is mathematically closely related to methods developed in the machine learning community for the analysis of text documents. Specifically, the model we fit can be regarded as a variant of a model used in probabilistic latent semantic analysis (PLSA)—a technique for automated detection of topics in a corpus of text—adapted to the present context of link communities. Connections between text analysis and community detection have been explored by several previous authors. Of particular interest is the work of Psorakis *et al.* [121], which, though it does not focus on link communities, uses another variant of the PLSA model, coupling it with an iterative fitting algorithm called nonnegative matrix factorization, to find overlapping communities in directed networks. Also of note is the work of Parkkinen *et al.* [115], who consider link communities as we do, but take a contrasting algorithmic approach based on a Bayesian generative model and Markov chain Monte Carlo techniques. A detailed description of the interesting connections between text processing and network analysis would take us some way from the primary purpose of this chapter, but for the interested reader we give a discussion and references in Appendix A.

2.4 Implementation

The method outlined above can be implemented directly as a computer algorithm for finding overlapping communities, and works well for networks of moderate size, up to tens of thousands of vertices. For larger networks both memory usage and run-time become substantial and prevent the application of the method to the largest systems, but both can be improved by using a more sophisticated implementation which makes applications to networks of millions of vertices possible.

The algorithm's memory use is determined by the space required to store the parameters: the θ_{iu} require $\mathcal{O}(nK)$ space while the $q_{ij}(u)$ require $\mathcal{O}(mK)$, where n and m are the numbers of vertices and edges in the network. Since m is usually substantially larger than n, this means that memory use is dominated by the $q_{ij}(u)$. We can reduce memory use by reorganizing the algorithm in such a way that the $q_{ij}(u)$ are never stored. Rather than focusing on the θ_{iu}, we work instead with the average number k_{iu} of ends of edges of color u connected to vertex i:

$$k_{iu} = \sum_j A_{ij} q_{ij}(u). \tag{2.9}$$

Given the values of these quantities on a given iteration of the algorithm, the calculation of the values at the next iteration is then as follows. First we define a new set of quantities k'_{iu} that will store the new values of the k_{iu}. Initially we set all of them to zero. We also calculate the average number κ_u of edges of color u summed over all vertices

$$\kappa_u = \sum_i k_{iu}, \tag{2.10}$$

in terms of which the original θ_{iu} parameters are

$$\theta_{iu} = \frac{k_{iu}}{\sqrt{\kappa_u}}, \tag{2.11}$$

where we have used Eq. (2.8). Next we go through each edge (i, j) in the network in turn and calculate the denominator of Eq. (2.5) for that i and j from the values of the k_{iu} thus:

$$D = \sum_u \theta_{iu}\theta_{ju} = \sum_u \frac{k_{iu}k_{ju}}{\kappa_u}. \tag{2.12}$$

Armed with this value we can calculate the value of $q_{ij}(u)$ for this i, j and all u from Eq. (2.5):

$$q_{ij}(u) = \frac{\theta_{iu}\theta_{ju}}{\sum_u \theta_{iu}\theta_{ju}} = \frac{k_{iu}k_{ju}}{D\kappa_u}. \tag{2.13}$$

Now we add this value onto the quantities k'_{iu} and k'_{ju}, discard the values of D and $q_{ij}(u)$, and repeat for the next edge in the network. When we have gone through all edges in this manner, the quantities k'_{iu} will be equal to the sum in Eq. (2.9), and hence will be the correct new values of k_{iu}.

This method requires us to store only the old and new values of k_{iu}, for a total of

$2nK$ quantities, and not the values of $q_{ij}(u)$. Depending on the values of m and n, this can result in substantial memory savings.

As for the running time, the algorithm as we have described it has a computational complexity of $\mathcal{O}(mK)$ operations per iteration of the equations, where m is again the number of edges in the network, but this too can be improved. In a typical application of the algorithm to a network, the end result is that each vertex belongs to only a subset of the K possible communities. To put that another way, we expect that many of the parameters k_{iu} will tend to zero under the EM iteration. It is straightforward to see from the equations above that if a particular k_{iu} ever becomes zero, then it must remain so for all future iterations, which means that it no longer need be updated and we can save ourselves time by excluding it from our calculations. This leads to two useful strategies for pruning our set of variables. In the first, we set to zero any k_{iu} that falls below a predetermined threshold δ. Once a k_{iu} has been set to zero, the corresponding values of the $q_{ij}(u)$ on all the adjacent edges are also zero and therefore need not be calculated. Thus, for each edge, we need only calculate the values of $q_{ij}(u)$ for those colors u for which both k_{iu} and k_{ju} are nonzero, i.e., for the intersection of the sets of colors at vertices i and j. This strategy leads to speed increases when the number of communities $K \gtrsim 4$. For smaller values of K the speed savings are outweighed by the additional computational overhead and it is more efficient to simply calculate all $q_{ij}(u)$, but we nonetheless still set the values of the k_{iu} to zero below the threshold δ because it makes possible our second pruning strategy.

Our second strategy, which can be used in tandem with the first and gives significant speed improvements for all values of K, is motivated by the observation that if all but one of the k_{iu} for a particular vertex are set to zero, then the color of the vertex, meaning the group or groups to which it belongs, is fixed at a single value and will no longer change at all. If both vertices at the ends of an edge (i, j) have this property, if both of them have converged to a single color and are no longer changing, then the edge connecting them no longer has any effect on the calculation and can be deleted entirely from the network.

By the use of these two strategies the speed of our calculations is improved markedly. We find in practice that the numbers of parameters k_{iu} and edges both shrink rapidly and substantially with the progress of the calculation, so that the majority of the iterations involve only a subset, typically those associated with the vertices whose community identification is most ambiguous. If the value of the threshold δ is set to zero, then the pruned algorithm is exactly equivalent to the original EM algorithm and the results are identical, yet even with this choice we find substantial speed improvements. If δ is chosen small but nonzero—we use $\delta = 0.001$ in our calculations—then we introduce an approximation into the calculation which means the results will be different in general from the original algorithm. In practice, however, the difference is small, and the nonzero δ gives us an additional and substantial speed improvement. (In our experiments we find a variation of about 1% or less in the final log-likelihood for values of δ anywhere from zero to 0.1. Note, however, that if the value of δ is greater than $1/K$, then it is possible inadvertently to prune all of the colors from a vertex, leaving it in no community at all. To avoid this, one must choose $\delta < 1/K$.)

A detailed comparison of results and run-times for the original and pruned versions of the algorithm is given in Appendix B for a range of networks. Unless stated otherwise, all calculations presented in the remainder of the chapter are done with the faster version of the algorithm.

2.5 Results

We have tested the performance of the algorithm described above using both synthetic (computer-generated) networks and a range of real-world examples. The synthetic networks allow us to test the algorithm's ability to detect known, planted community structure under controlled conditions, while the real networks allow us to observe performance under practical, real-world conditions.

2.5.1 Synthetic networks

Our synthetic network examples take the form of a classic consistency test. We generate networks using the same stochastic model that the algorithm itself is based on and measure the algorithm's ability to recover the known community divisions for various values of the parameters. One can vary the values to create networks with stark community structure (which should make detection easy) or no community structure at all (which makes it impossible), and everything in between, and we can thereby vary the difficulty of the challenge we pose to the algorithm.

The networks we use for our tests have $n = 10\,000$ vertices each, divided into two overlapping communities. We place x vertices in the first community only, meaning they have connections only to others in that community, y vertices in the second community only, and the remaining $z = n - x - y$ vertices in both communities, with equal numbers of connections to vertices in either group on average. We fix the expected degree of all vertices to take the same value k.

We perform three sets of tests. In the first we fix the size of the overlap between the communities at $z = 500$, divide the remaining vertices evenly $x = y = 4750$, and observe the behavior of the algorithm as we vary the value of k. When $k \to 0$ there are no edges in the network and hence no community structure, and we expect the algorithm (or any algorithm) to fail. When k is large, on the other hand, it should be straightforward to work out where the communities are.

For our second set of tests we again set the overlap at $z = 500$ but this time we fix $k = 10$ and vary the balance of vertices between x and y. Finally, for our third set of tests we set $k = 10$ and constrain x and y to be equal, but allow the size z of the overlap to vary.

In Fig. 2.1 we show the measured fraction of vertices classified correctly (black curve) in each of these three sets of tests (the three separate panels), averaged over 100 networks for each point. To be considered correctly classified a vertex's membership (or lack of membership) in both groups must be reported correctly by the algorithm, and the algorithm considers any vertex to be a member of a group if, on average, it has at least one edge of the appropriate color when the maximum-likelihood fitting procedure is complete. In mathematical terms, a vertex belongs to community u if its expected degree with respect to color u, given by $\sum_j A_{ij}q_{ij}(u)$, is greater than one.

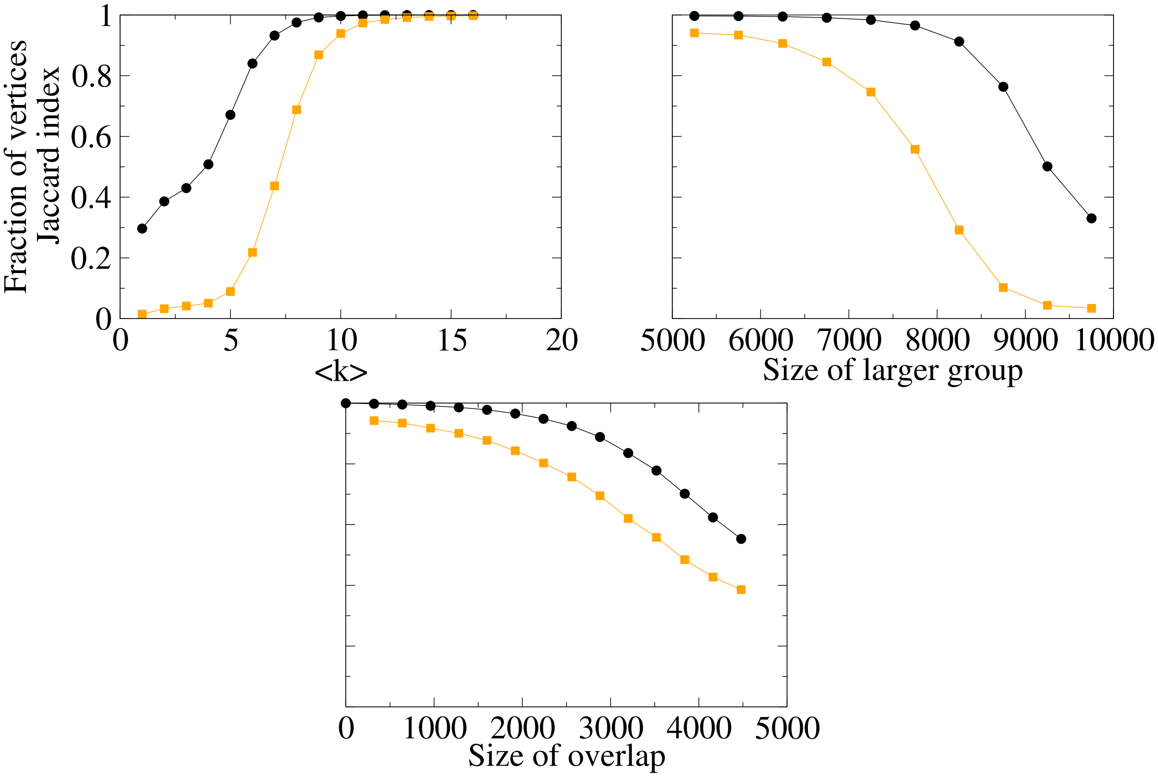


Figure 2.1: Results from the three sets of synthetic tests described in the text. Each data point is averaged over 100 networks. Twenty random initializations of the variables were used for each network and the run giving the highest value of the log-likelihood was taken as the final result. In each panel the black curve shows the fraction of vertices assigned to the correct communities by the algorithm, while the lighter curve is the Jaccard index for the vertices in the overlap. Error bars are smaller than the points in all cases.

As the figure shows, there are substantial parameter ranges for all three tests for which the algorithm performs well, correctly classifying most of the vertices in the network. As expected the accuracy in the first test increases with increasing k and for values of k greater than about ten—a figure easily attained by many real-world networks—the algorithm identifies the known community structure essentially perfectly. In the other two tests accuracy declines as either the asymmetry of the two groups or the size of the overlap increases, but approaches 100% when either is small.

To probe in more detail the algorithm's ability to identify overlapping communities, we have also measured, for the same test networks, a Jaccard index: if O^{true} is the set of vertices in the true overlap and O^{found} is the set the algorithm identifies as being in the overlap, then the Jaccard index is $J = |O^{true} \cap O^{found}|/|O^{true} \cup O^{found}|$. This index is a standard measure of similarity between sets that rewards accurate identification of the overlap while penalizing both false positives and false negatives. The values of the index are shown as the lighter curves in Fig. 2.1 and, as we can see, the general shape of the curves is similar to the overall fraction of correctly identified vertices. In particular, we note that for networks with sufficiently high average degree k the value of J tends to 1, implying that the overlap is identified essentially perfectly.

2.5.2 Real networks

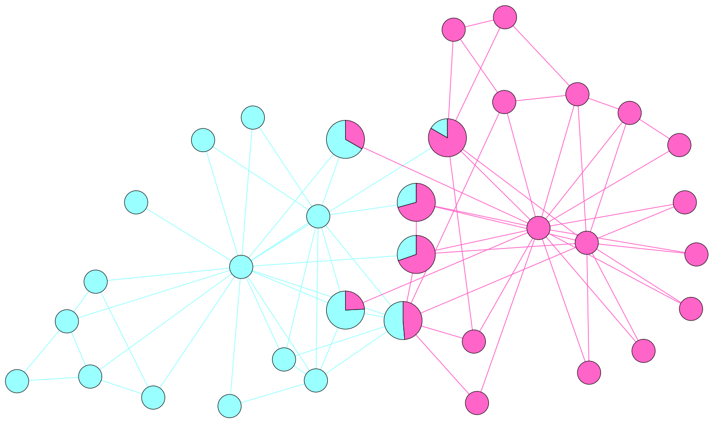
We have also tested our method on numerous real-world networks. In this section we give detailed results for four specific examples. Summary results for a number of additional examples are given in Appendix B.

Our first example is one that has become virtually obligatory in tests of community detection, Zachary's "karate club" network, which represents friendship patterns between members of a university sports club, deduced from an observational study [152]. The network is interesting because the club split in two during the study, as a result of an internal dispute, and it has been found repeatedly that one can deduce the lines of the split from a knowledge of the network structure alone [54, 58].

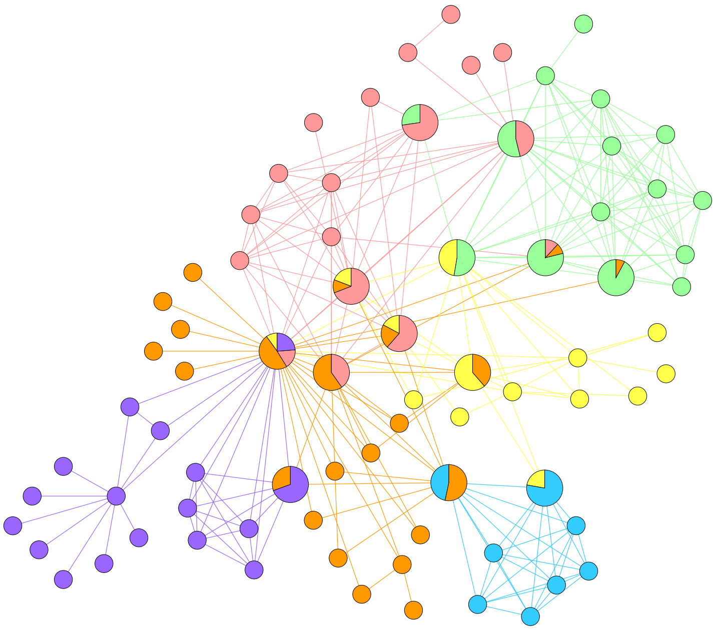
Figure 2.2a shows the decomposition of the karate club network into two overlapping groups as found by our algorithm. The colors in the figure show both the division of the vertices and the division of the edges. The split between the two groups in the club is clearly evident in the results and corresponds well with the acknowledged "ground truth," but in addition the algorithm assigns several vertices to both groups. The individuals represented by these overlap vertices, being by definition those who have friends in both camps, might be supposed to have had some difficulty deciding which side of the dispute to come down on, and indeed Zachary's original discussion of the split includes some indications that this was the case [152]. Note also that, in addition to identifying overlapping vertices, our method can assign to each a fraction by which it belongs to one community or the other, represented in the figure by the pie-chart coloring of the vertices in the overlap. The fraction is calculated as the expected fraction of edges of each color incident on the vertex.

Our second example is another social network and again one whose community structure has been studied previously. This network, compiled by Knuth [85], represents the patterns of interactions between the fictional characters in the novel *Les Misérables* by Victor Hugo. In this network two characters are connected by an edge if they appear in the same chapter of the book. Figure 2.2b shows our algorithm's partition of the network into six overlapping communities and the partition accords roughly with social divisions and subplots in the plot-line of the novel. But what is particularly interesting in this case is the role played by the hubs in the network—the major characters who are represented by vertices of especially high degree. It is common to find high-degree hubs in networks of many kinds, vertices with so many connections that they have links to every part of the network, and their presence causes problems for traditional, nonoverlapping community detection schemes because they do not fit comfortably in any community: no matter where we place a hub it is going to have many connections to vertices in other communities. Overlapping communities provide an elegant solution to this problem because we can place the hubs in the overlaps. As Fig. 2.2b shows, our algorithm does exactly this, placing many of the hubs in the network in two or more communities. Such an assignment is in this case also realistic in terms of the plot of the novel: the major characters represented by the hubs are precisely those that appear in more than one of the book's subplots.

A similar behavior can be seen in our third example, which is a transportation network, the network of passenger airline flights between airports in the United States. In this network, based on data for flights in 2004, the vertices represent airports and an edge between airports indicates a regular scheduled direct flight. Spatial networks, those in which, as here, the vertices have well-defined positions in geographic space, are often found to have higher probability of connection for vertex pairs located closer together [14, 56], which suggests that communities, if they exist, should be regional, consisting principally of blocks of nearby vertices. The communities detected by our algorithm in the airline network follow this pattern, as shown in Fig. 2.3. The three-way split shown divides the network into east and west coast groups and a group for Alaska. The overlaps are composed partly of vertices that lie along the geographic boundaries between the groups, but again include hubs as well, which tend to be placed in the overlaps even when they don't lie on boundaries. As with the previous example, this placement gives the algorithm a solution to the otherwise difficult problem of assigning to any one group a hub that has connections to all parts of the network. But it also makes intuitive sense. Hubs are the "brokers" of the airline network, the vertices that connect different communities together, since they are precisely the airports that passengers pass through in traveling between distant locations. Thus it is appropriate that hubs be considered members of more than one



(a)



(b)

Figure 2.2: Overlapping communities in (a) the karate club network of [152] and (b) the network of characters from *Les Misérables* [85], as calculated using the algorithm described in this chapter. The edge colors correspond to the highest value of $q_{ij}(u)$ for the given edge, while vertex colors indicate the fraction of incident edges that fall in each community. For vertices in more than one community the vertices are drawn larger for clarity and divided into pie charts representing their division among communities.

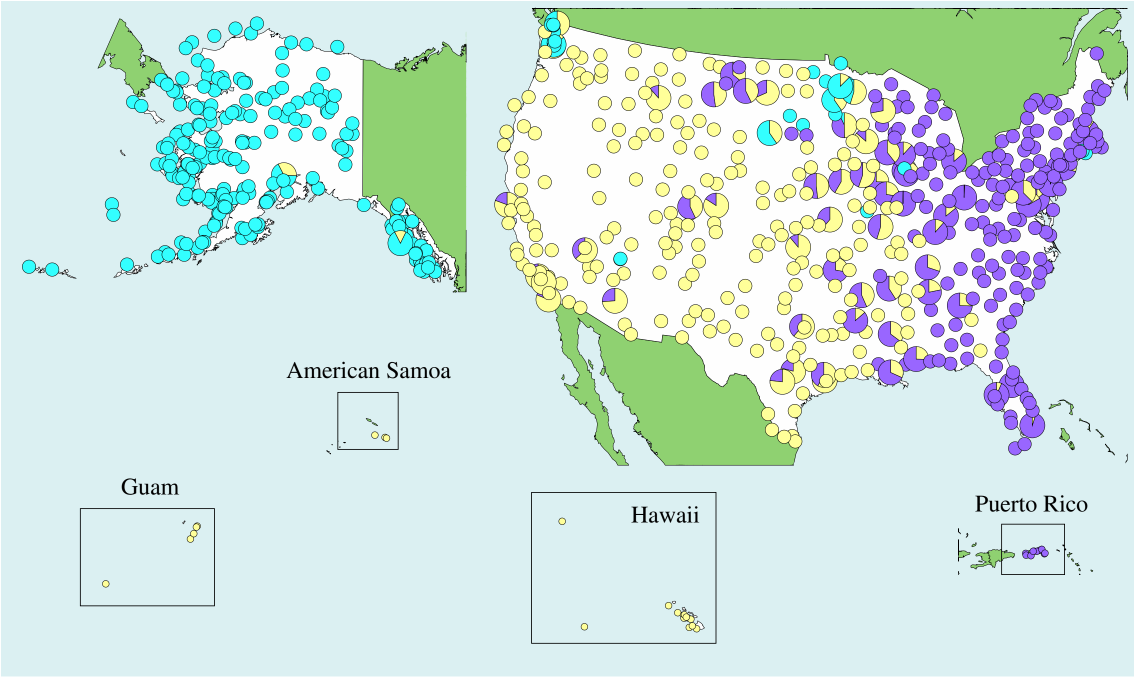


Figure 2.3: Overlapping communities in the network of US passenger air transportation. The three communities produced by the calculation correspond roughly to the east and west coasts of the country and Alaska.

group. In most cases the hubs belong most strongly to the community in which they are geographically located, and less strongly to other communities.

For our fourth example we examine a network of coauthorships between researchers publishing on network science. In this network, which was previously published in [103], vertices represent scientists and unweighted edges connect pairs of scientists who have coauthored at least one paper together. Figure 2.4a shows the division of the network's largest component as found by our algorithm for $K = 12$ communities.

The figure reveals a new phenomenon not present in our previous examples: some of the communities found by the algorithm are not contiguous—they are divided into two or more separate parts with no edges connecting the parts. This seems unsatisfactory. Intuitively, one expects communities to be connected.

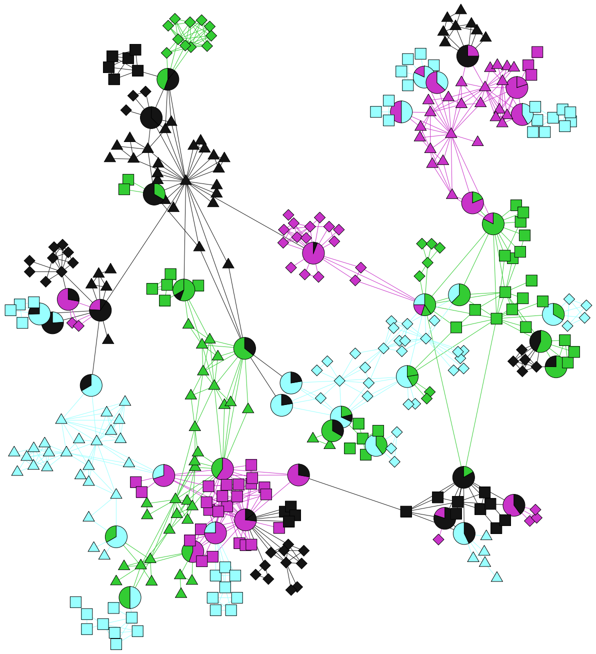
The explanation for this behavior is that in this case the algorithm has found a local optimum of the likelihood, rather than a global one, and the local optimum contains disconnected communities. To address this issue, we adopt the following procedure. After the communities are calculated with the EM algorithm we find all their connected clusters, then work through them in order from smallest to largest. Each cluster is added to the neighboring cluster (of any community) with which it has the most connections, unless it is the only cluster in its community, in which case we reverse the process and add the neighboring cluster to it. The only exception is when all neighboring clusters are the only cluster in their community, in which case we do nothing. Then we move on to the next largest cluster, bearing in mind that cluster sizes may have changed in the process. When we have gone through all clusters in this manner we are left with K communities, each of which is connected, consisting of a single cluster, and any connected pair of vertices that were originally assigned to the same cluster by the EM algorithm will still be in the same cluster.

This procedure requires very little additional effort to perform and in our experiments we find that it always increases the likelihood of the community assignment, indicating that indeed the original EM algorithm found a local likelihood maximum. Figure 2.4b shows the result of applying the procedure to our coauthorship network and, as the figure shows, the communities found are now connected[3].

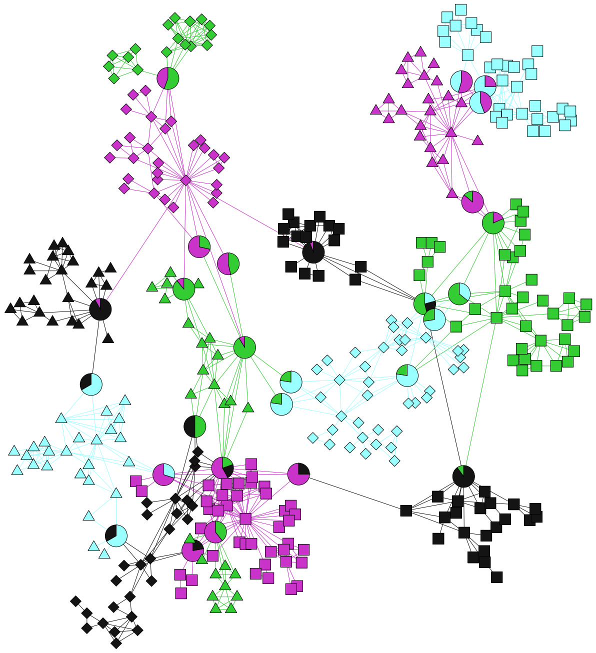
2.6 Nonoverlapping communities

As we have described it, our algorithm is an algorithm for finding overlapping communities in networks, but it can be used to find nonoverlapping communities as well. As pointed out by a number of previous authors [121, 141, 153], any algorithm that calculates proportional membership of vertices in communities can be adapted to the nonoverlapping case by assigning each vertex to the single community to which it belongs most strongly. In our case, this means assigning vertices to the community for

[3] We have also applied the same procedure to the previous example networks, including the synthetically generated ones, but it produced in no significant changes to the results in those cases.



(a)



(b)

Figure 2.4: Overlapping communities in the collaboration network of network scientists as calculated by the algorithm of Section 2.4 (a) without the post-processing step that ensures connected communities and (b) with the post-processing. Each community is represented as a shape/color combination, except for overlapping vertices, which are always drawn as circles.

which the value of k_{iu}/κ_u is largest. It turns out that this procedure can be justified rigorously in our case by regarding the link community model as a relaxation of a nonoverlapping degree-corrected stochastic blockmodel. The details are given in Appendix C. Here we give some example applications to show how the approach works in practice.

As with the overlapping case, we test the method on both synthetic and real-world networks. For the synthetic case we use a standard test, the LFR benchmark for unweighted undirected networks with planted community structure [86, 87]. To make possible comparisons with the previous study of Ref. [86] we use the same parameters, with networks of 1000 and 5000 vertices, average degree 20, maximum degree 50, degree exponent -2, and community exponent -1. We also use the same two ranges of community sizes, with communities of 10 to 50 vertices for one set of tests (labeled S for "small" in our figures) and 20 to 100 vertices for the other set (labeled B for "big"). The value of K for the detection algorithm was set equal to the number of communities in the benchmark network (which, because of the nature of the benchmark, is not a constant but varies from one network to another).

To quantify our algorithm's success at detecting the known communities in the benchmark networks we use the variant normalized mutual information measure proposed in [86]. We note that this measure is different, and in general returns different results, from the normalized mutual information measure most often used to evaluate community structure [33], but using it allows us to make direct comparisons with the results for other algorithms given in [86].

In our benchmark tests we find that the method described above for finding nonoverlapping communities—just choosing the community with the highest value of k_{iu}/κ_u—returns only average performance when compared with the other algorithms tested in Ref. [86]. However, a simple modification of the algorithm produces significantly better results: after generating a candidate division into communities using the rounding method, we then apply a further optimization step in which we move from one community to another the single vertex that most increases the log-likelihood of the division under the stochastic blockmodel, and repeat this exercise until no further such moves exist. This process, which is reminiscent of the well-known Kernighan–Lin algorithm for graph partitioning [84], is easy to implement and carries little computational cost when compared to the calculation of the initial division, but it improves our results dramatically.

The results of our tests are shown in Figure 2.5. The top panel shows the performance of the algorithm without the additional optimization step and the results fall in the middle of the pack when compared to previous algorithms, better than some methods but not as good as others. The bottom panel shows the results with the additional optimization step added, and now the algorithm performs about as well as, or better than, the algorithms analyzed in Ref. [86]. The general shape of the mutual information curve is similar to that of the best competing methods, falling off around the same place, although the mutual information values are somewhat lower for low values of the mixing parameter, indicating that the method is not getting the community structure exactly correct in this regime. Examining the communities in detail reveals that the method occasionally splits or merges communities. It is possible that performance could be improved further by a less simple-minded post-processing step for optimizing the likelihood. In particular, by contrast with the overlapping groups of the previous section, we made no effort to ensure that the communities in the present tests consisted of only a single cluster, and doing so might potentially improve the results.

We also give, in Fig. 2.6, an example of a test of the method against a real-world network, in this case the much studied college football network of Ref. [58]. In this network the vertices represent university teams in American football and the edges represent the schedule of games for the year 2000 football season, two teams being

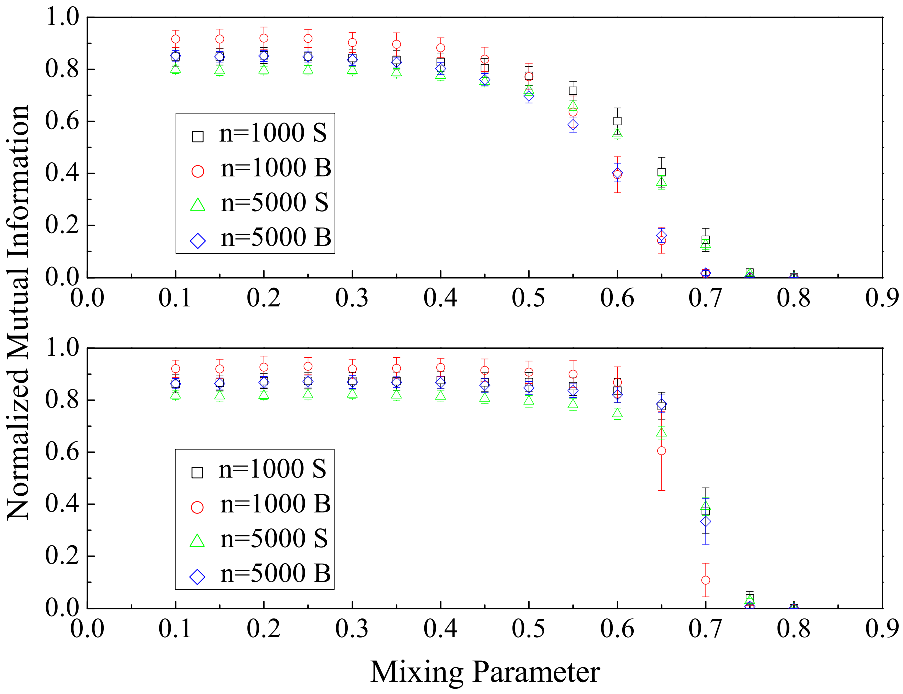


Figure 2.5: Performance of the nonoverlapping community algorithm described in the text when applied to synthetic networks generated using the LFR benchmark model of Lancichinetti *et al.* [86]. Parameters used are the same as in Ref. [86] and (S) and (B) denote networks with the "small" and "big" community sizes used by the same authors. The top and bottom panels respectively show the results without and with post-processing to optimize the value of the log-likelihood. Ten random initializations of the variables were used for each network and each point is an average over 100 networks.

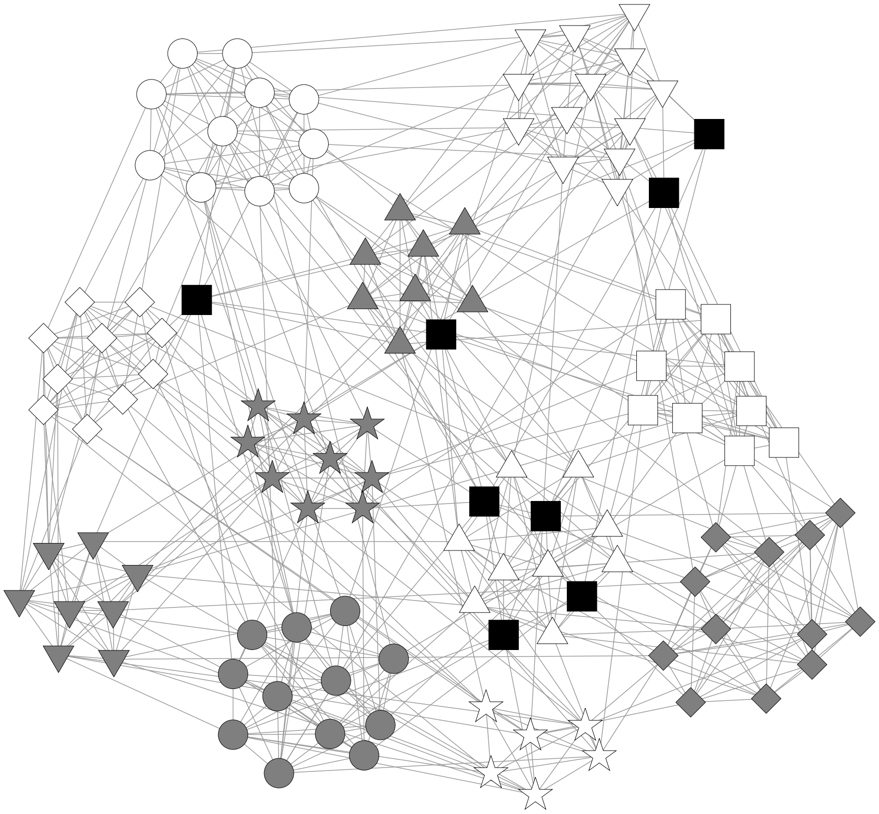


Figure 2.6: Non-overlapping communities found in the US college football network of Ref. [58]. The clusters of vertices represent the communities found by the algorithm, while the vertex shape and color combination represents the "conferences" into which the colleges are formally divided. As we can see, the algorithm in this case extracts the known conference structure perfectly. (The square black vertices represent independent colleges that belong to no conference.)

connected if they played a game. It has been found in repeated analyses that a clustering of this network into communities can retrieve the organizational units of US college sports, called "conferences," into which universities are divided for the purposes of competition. In 2000 there were 11 conferences among the Division I-A teams that make up the network, as well as 8 teams independent of any conference. As Fig. 2.6 shows, every single team that belongs in a conference is placed correctly by our algorithm.

2.7 Discussion

In this chapter we have described a method for detecting communities, either overlapping or not, in undirected networks. The method has a rigorous mathematical foundation, being based on a probabilistic model of link communities; is easy to implement, fast enough for networks of millions of vertices; and gives results competitive with other algorithms.

Nonetheless, the method is not perfect. Its main current drawback is that it offers no criterion for determining the value of the parameter we call K, the number of communities in a network. This is a perennial problem for community detection methods of all kinds. Some methods, such as modularity maximization, do offer a solution to the problem, but that solution is known to give biased answers or be inconsistent under at least some circumstances [17, 55]. More rigorous approaches such as the Bayesian information criterion [129] and the Akaike information criterion [5] are unfortunately not applicable here because many of the model parameters are zero, putting them on the boundary of the parameter space, which invalidates the assumptions made in deriving these criteria.

Another approach to choosing the value of K is to perform the calculations with a large value and regularize the parameters in a manner such that some communities disappear, meaning that zero edges are associated with those communities. For example, Psorakis *et al.* [121], in studies using their matrix factorization algorithm, used priors that penalized their model for including too many nonzero parameter values and hence created a balance between numbers of communities and goodness of fit to the network data. Unfortunately, the priors themselves contain undetermined parameters whose values can influence the number of communities and hence the problem is not completely solved by this approach.

We believe that statistical model selection methods applied to generative models should in principle be able to find the number of communities in a consistent and satisfactory manner. In the next chapter, we create a heuristic based on likelihood ratio tests as a first attempt at this problem that does not rely on slow numerical methods like bootstrapping.

CHAPTER III

Number of communities in an overlapping communities model

3.1 Introduction

Many networks, regardless of origin, exhibit some sort of grouping phenomenon. Subsets of vertices have many connections within the subset, and fewer connections between the subsets [54, 58]. Finding these groups, a process known as community detection, is a fundamental problem in the study of networks. There are a wide variety of methods used to detect communities in networks, from local aggregation methods that only take a vertex's immediate connections into account to global quality functions and statistical network models that use the entire network structure to fit the communities [108, 112, 131].

As mentioned in chapter II, one very important question regarding community detection that has not received very much attention is deciding how many communities are in the network. Most global methods require a fixed number of communities as an input, and no principled way to directly compare the fits between different numbers of communities. The statistics literature has several methods to decide which model taken from a set of given models gives the best fit for a dataset [5, 129, 149]. These methods could potentially be adapted and applied to the statistical network models, and in a few cases have been used [18, 151]. However, there has been little effort to use these methods to find the correct number of communities [18], and much of the work that has been done did not study the methods in a rigorous manner [121].

In this chapter, we provide a heuristic method for choosing the number of communities in the overlapping community detection blockmodel defined in chapter II, originally presented in [10]. This heuristic is based on likelihood ratio tests [149]. This provides us with an effective, quick way of determining if a new community is significant or not. By iterating the test for incrementing numbers of communities, we find the best number of communities for the network.

The chapter is organized as follows. First we restate the definition of the community detection model that we will be focusing on, how it is used to extract communities from a network, and an interpretation for the types of structure the model finds. Then we give our heuristic test for selecting the number of communities for the model and discuss its relation to likelihood ratio tests from the statistics literature. Finally we give both synthetic and real-world examples demonstrating the application of this test and show that it outperforms standard statistical methods.

3.2 Overlapping community detection model recap

Since the overlapping community detection blockmodel is central to this chapter, it's worth discussing its formulation for K communities [10]. The basic premise is that each vertex has a K-dimensional group vector $\theta_{\mathbf{i}}$ that controls its community membership as well as its degree distribution in each of those communities. This model generates link communities, with the expected number of edges associated with community u between vertices i and j is $\theta_{iu}\theta_{ju}$. When extracting communities using this model however, we don't know the communities for each edge, observing only the presence or lack of edges, so we have to treat the link communities as latent variables and sum over them. Thus the expected number of edges between each pair of vertices can be thought of as the inner product of the community vectors, $\langle A_{ij}\rangle = \sum_u \theta_{iu}\theta_{ju}$. We use a Poisson formulation for the edges in each community so that A_{ij} is also Poisson distributed, so technically we can have multi-edges, although generally the number of these is small since we're primarily interested in sparse networks. The full likelihood for this undirected network model is

$$\begin{aligned} P(G|\mathbf{\Theta};K) = \prod_{i<j} \frac{\bigl(\sum_u \theta_{iu}\theta_{ju}\bigr)^{A_{ij}}}{A_{ij}!} \exp\Bigl(-\sum_u \theta_{iu}\theta_{ju}\Bigr) \\ \times \prod_i \frac{\bigl(\frac{1}{2}\sum_u \theta_{iu}\theta_{iu}\bigr)^{A_{ii}/2}}{(A_{ii}/2)!} \exp\Bigl(-\tfrac{1}{2}\sum_u \theta_{iu}\theta_{iu}\Bigr). \end{aligned} \tag{3.1}$$

Thus the log-likelihood is

$$\begin{aligned} \log P(G|\mathbf{\Theta};K) = \tfrac{1}{2}\Biggl(\sum_{ij} A_{ij}\log\bigl(\textstyle\sum_u \theta_{iu}\theta_{ju}\bigr) - \displaystyle\sum_{iju}\theta_{iu}\theta_{ju}\Biggr) \\ - \sum_{i<j}\log(A_{ij}!) - \sum_i \log\bigl((A_{ii}/2)!\bigr) - \sum_i A_{ii}\log(2)/2. \end{aligned} \tag{3.2}$$

Although many of these terms are constants with respect to the network and thus unnecessary for finding the best fit, we include them for completeness. Furthermore, the multiplicative constants are necessary for computing our heuristic.

To extract communities from a network using this model, we apply an expectation-maximization (EM) algorithm and obtain a closed form iterative solution, the details of which are shown in section 2.3. Computing equations (3.3) and (3.4) alternately until they converge will give the desired best fit. The $q_{ij}(u)$ correspond to the latent communities affiliated with each edge.

$$q_{ij}(u) = \frac{\theta_{iu}\theta_{ju}}{\sum_u \theta_{iu}\theta_{ju}} \tag{3.3}$$

$$\theta_{iu} = \frac{\sum_j A_{ij} q_{ij}(u)}{\sqrt{\sum_{ij} A_{ij} q_{ij}(u)}} \tag{3.4}$$

This overlapping community detection blockmodel attempts to find communities that are connected to each other through vertices rather than edges. In other words, vertices only connect to other vertices in the same community (or in the overlap with that community), and the vertices in multiple communities are located on the boundary between their associated communities, acting as a bridge between the communities. This is a consequence of the fact that this blockmodel does not have a mixing matrix – connections between vertices with no shared communities are explicitly forbidden. Looking at the blockmodel from a different perspective, when given the community parameters, it is a superposition of Chung-Lu random graphs [28]. The generation of an edge between two vertices that share only one community is completely independent of the other communities. At some level (although in a rigorous sense this isn't what is going on), the best fit of a network to the model is one that makes the individual community subgraphs as independent as possible – that is, minimizes the size of the overlap with respect to edges and communities.

3.3 Counting heuristic

With the previous section in mind, suppose we have a network where there are K communities and we wish to extract $K + 1$ communities from it, intentionally overfitting the network. Based on empirical evidence, you really have two options for the best community structure: either you can take one of the K communities and break it into two new communities, or you can take one of the overlaps between two communities in the K sense and force it to become its own new community, and in either situation leave the remaining $K - 1$ communities as they are. The latter is a special circumstance and only occurs when a fairly dense community that doesn't have any additional community structure was somehow previously split into two groups, which forces a large fraction of the vertices to be in the overlap[1]. However, the latter case is in contradiction with the initial assumption that there really were K communities, so we disregard this case.

Define the set $C_y(K)$ to be the yth community out of K communities in the network, as fit by our overlapping community detection blockmodel. This set should include overlapping vertices that are in different communities in addition to y (mathematically, the set of $C_y(K)$ for fixed K forms a cover of the network). Suppose we have the situation from the previous paragraph: we take a network with K communities in it and extract $K+1$ communities. Following the logic from the previous paragraph, let's say without loss of generality that the community $C_1(K)$ is split into the communities $C_1(K+1)$ and $C_{K+1}(K+1)$. Call the set of communities vertex i is in when fitting K communities $V_i(K)$. Then we count the number of the vertices c that satisfy either of the following conditions:

List 1.

1. $V_i(K) = \{1\}$

2. $|V_i(K+1)| \geq |V_i(K)| > 1$ and $\{1, K+1\} \cap V_i(K+1) \neq \emptyset$.

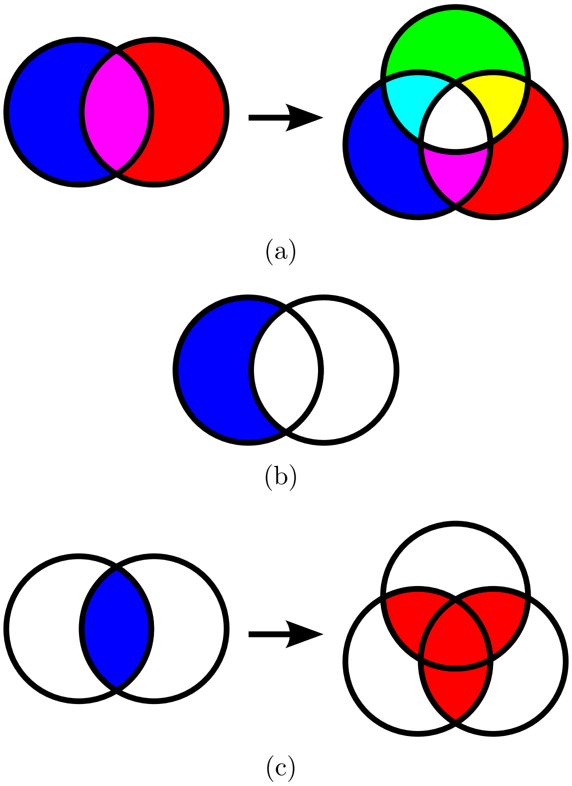
In other words, a vertex that is entirely in the split community gets counted, as do overlapping vertices in the K community case that are in at least as many groups in the $K+1$ case, where one of those groups is either of the groups that were split off. Notice in particular that this doesn't require such a vertex to share communities with the original group $C_1(K)$ – ideally this condition would also be required, but since the assumption of communities splitting is at best an approximation, it is actually

[1] In the case where a community with no additional community structure is split into two subcommunities, as the degree of a vertex increases, the probability that it will have an edge to a vertex in the other community grows exponentially. Make the entire community dense enough and the majority of vertices will have edges to both communities. This is mitigated by combinatorics for the possible sets of communities, but only to a certain extent [63].

better to count the additional vertices. An example of how to count the vertices in a network with two communities is shown in figure 3.1. Then we approximate that the distribution of twice the change in log-likelihood between K communities and $K+1$ communities to be χ^2 distributed with parameter defined by twice this count: $2\Delta\mathcal{L} \sim \chi^2(2c)$.

The final use for the heuristic with regards to the overlapping community detection blockmodel involves extracting communities in an incremental fashion using the EM algorithm as described in section 3.2, comparing consecutive K by computing p-values using the method above until the p-value falls below a given threshold. Each iteration, we decide which group is split into two by counting the number of edges that were in each of the K communities that are now in each of the $K+1$ communities, assuming no *a priori* choice when the edges are in multiple groups. The group from the K community extraction that split is the one where 2 of the $K+1$ communities primarily mapped to it with respect to the edges. The rest is straightforward counting and then a direct computation of the p-value and running the EM algorithm for varying K. The final output is both the number of communities as well as the community memberships themselves.

Note that the heuristic method assumes that the absolute maximum is found at each value of K, since the community structure found is assumed to be the true structure for the $K+1$ test case. Thankfully, it was demonstrated in [10] that the EM algorithm does a good job of finding planted community structure. While this dependence on the previously fit community structure may seem like a weakness of the method, we will demonstrate in section 3.5 that it is actually a necessary feature of the method since the expected change in likelihood does depend on the actual (and hence extracted) community structure.


(a)

(b)

(c)

Figure 3.1: Communities in a network demonstrated in the form of a Venn diagram. The test is to assume 2 communities are in the network and extract 3 communities. (a) The community structure. From left to right, the blue community was split into blue and green communities. Colors are additive for overlaps – for example, a vertex in the red and blue communities falls in the purple segment. (b) The first case from list 1. Any vertices belonging to the community set highlighted in blue count for the heuristic. (c) The second case from list 1. Any vertices in the community set highlighted in blue that are extracted as belonging to any of the red sets are counted. More complicated situations can arise when 3 or more communities are present in the network.

3.4 Relation to Likelihood Ratio Tests

Our heuristic method bears resemblance to likelihood ratio tests. In particular, the well-known Wilks' Theorem has a relation to our method, combined with intuition where the theorem itself breaks down [149]. It is well-known that simple model selection methods break down for models with large numbers of parameters (in particular when the number of parameters is a function of the size of the dataset), but much work has been done to extend the methods to this case [51, 79, 137]. Much of the work was done by introducing a penalty term to reduce the effective number of parameters, which allows us to more easily select the correct model due to the reduced variance in the resulting distributions. As we shall see, fits to our model naturally reduce the effective number of parameters, so we can gain insight from the literature.

Wilks' Theorem states that twice the difference in the maximum log-likelihood between two nested models is χ^2 distributed with mean equal to the difference in the number of parameters between the two models. For the overlapping community detection blockmodel, the difference in the number of parameters between K and $K + 1$ communities is exactly n since each vertex has a single new θ parameter. The general Wilks' Theorem is reliant upon the model being smooth with respect to the parameters and that the maximum can be attained[2]. These both seem fairly straightforward in the case of our blockmodel (and in fact we prove the latter in Appendix D – the former is trivial on the interior of the space, where $\theta_{iz} > 0$). However, as shown by Chen *et al.*, the theorem can also have problems when degenerate or redundant parameters exist, for example at the boundaries of the parameter space, when multiple parameters can describe the same effects in the model, or when a pa-

[2] While this is not necessary for the mean, the shape of the distribution (χ^2) also relies on the common assumption that the data set is large so that the central limit theorem applies. This does not apply for networks, where we have one observation of the network, but the distribution is often still χ^2 because of the fact Poisson distributions converge quickly to Gaussian distributions as the parameter increases.

rameter is unidentifiable due to another parameter effectively removing it from the model [25, 26]. The overlapping community detection blockmodel fits these criteria – it is very common for parameters to go to 0, indicating that the corresponding vertex is not at all in that particular community. Between this effective parameter reduction and the extensive number of parameters we can't directly apply Wilks' Theorem to our problem of finding the number of communities in a network with this model.

It seems logical that any degenerate parameters should not contribute to the change in log-likelihood. It is with this in mind that we created our heuristic. When we split a community into two communities, the only vertices that we count are the ones that need the additional parameter to fit the new solution – vertices that aren't in the community that split certainly didn't need the additional parameter. The additional multiplicative factor of two compared to Wilks' Theorem is based on observation of the difference between the logical prediction and observation.

It is not clear where this additional factor of two comes from. One possibility is that it comes from the undirected nature of the network. However, since we have properly computed the full log-likelihood as in equation (3.2), it would have to be a non-trivial dependence. Another possibility is that it could have to do with biased versus unbiased estimators, similar to the difference between Wilks' Theorem and the Akaike Information Criterion [5].

3.5 Results

In this section we show results of our counting method on both synthetic networks as well as real networks. We show that as we should expect, the generic methods do a poor job predicting the distributions for the change in log-likelihood even in networks that were generated using the overlapping community detection blockmodel. However, the new heuristic does a good job of predicting the distributions associated with the change in log-likelihood and thus can be used to accurately identify the number of communities in the network.

To demonstrate that our heuristic works well on synthetic networks where other model selection methods fail, we generated networks with 999 vertices and an expected degree of 20 for each vertex. There are 3 communities of equal sizes, and each vertex has on average 80% of its connections to other vertices in the same group, with the remaining 20% being split evenly between vertices in the other two communities. The parameters were chosen so that the EM algorithm could easily detect the planted community structure, and we ran 1000 individual networks with this parameter set to get an accurate estimation of the change in log-likelihood distributions. In each network, we computed our heuristic's prediction for the change in log-likelihood, but the final stopping criterion was Wilks' Theorem to demonstrate each method's success and failure, since that was the least stringent test.

The results from our experiment are shown in figure 3.2. For $K = 2, 3$, the observed change in log-likelihood (normalized by n) is significantly larger than what any of our tests predicted. This should be the case, since there's additional community structure in the network whereas our tests assume none. As is expected and can be seen, Wilks' Theorem overestimates the number of communities present in the network, most commonly selecting 5 communities but occasionally choosing 4 or 6. On the other hand, our counting heuristic matches up well with $2\Delta\mathcal{L}$ starting at $K = 4$ as you would expect (since $K = 4$ should be the first value to fail the test given that there are only 3 communities in the network) and continues to make accurate predictions beyond that point, and thus would also predict the number of communities. The Akaike Information Criterion (AIC) [5], another common statistical model selection method (that like Wilks' Theorem is intentionally erroneously used here) correctly identifies the number of communities in the network. AIC introduces a penalty to the log-likelihood equal to the number of parameters, and the resulting expression is maximized over the parameters and the number of parameters. This

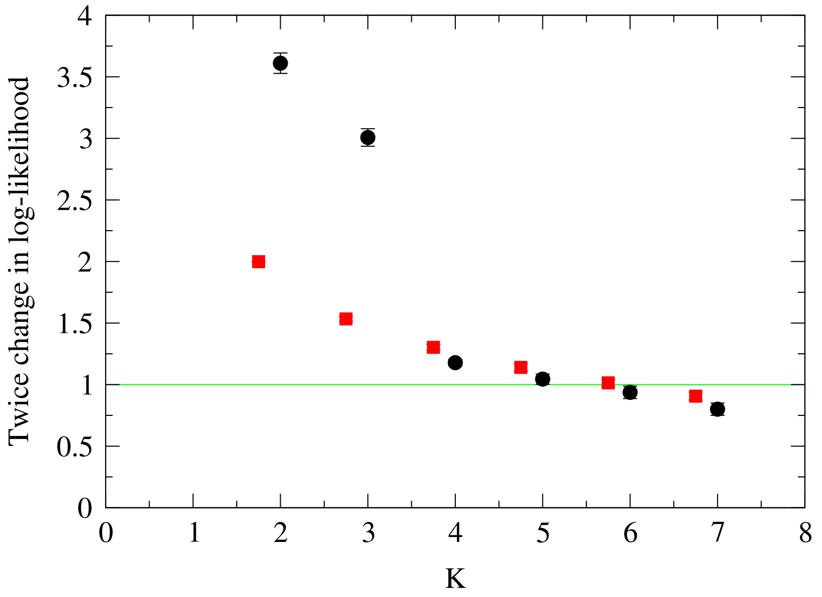


Figure 3.2: Twice the change in log-likelihood normalized by the number of vertices between K communities and $K-1$ communities plotted against K. The stopping point for each network is when it falls below the green line (Wilks' Theorem prediction). Black circles are the average and standard deviation of the actual data, and red squares are our prediction based on the heuristic described in the text. AIC's stopping criterion is not drawn, but would be a horizontal line at 2.0.

can be interpreted like Wilks' Theorem, comparing the log-likelihoods of consecutive K against the number of parameters until the maximum is found. According to AIC, in the lack of additional structure, twice the change in log-likelihood should be $2n$ for our overlapping community detection model (and in general twice the mean of what Wilks' Theorem predicts, albeit with no variance), but this estimation of the change in log-likelihood is very inaccurate, so we expect that it should fail on networks with more complicated structure, and we will demonstrate this later in this section.

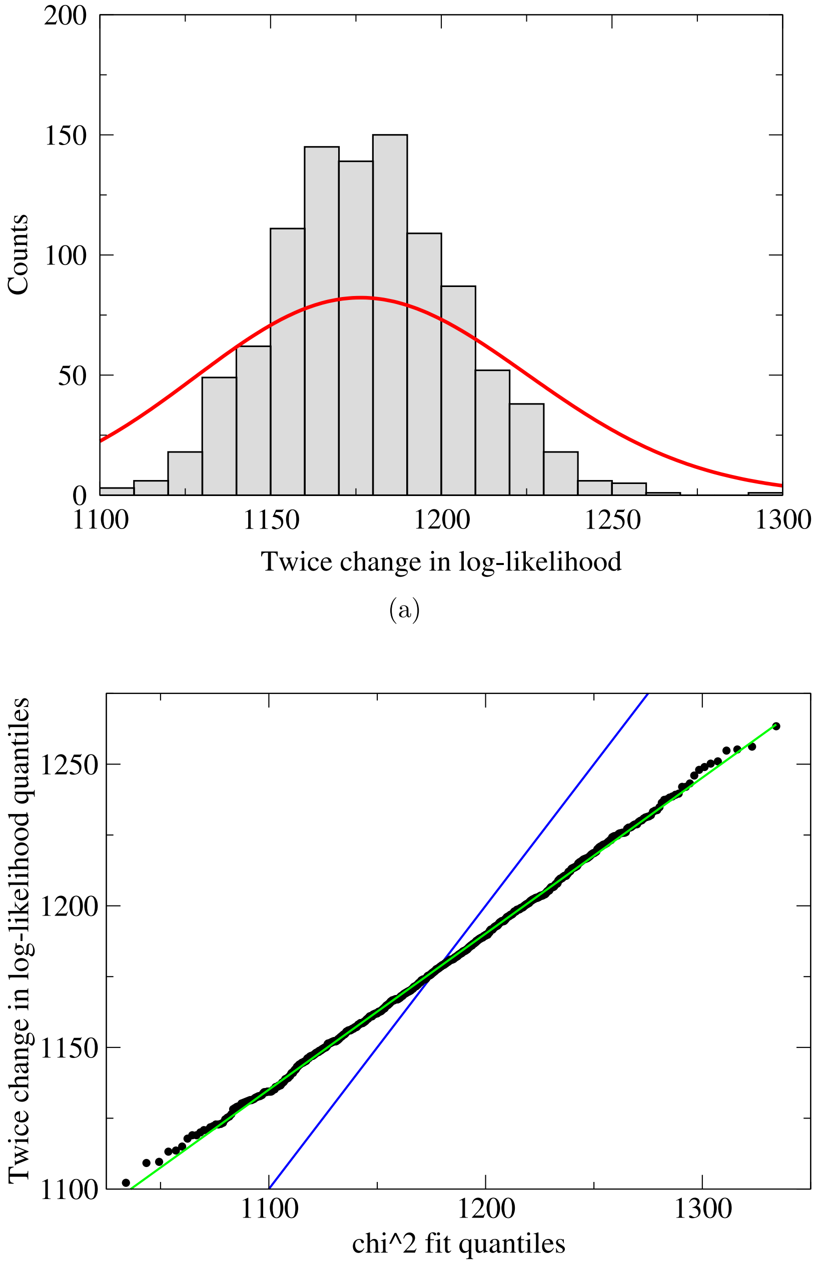
We can examine the change in log-likelihood at each K in further detail to compare the distributions our heuristic expects versus the observed distribution for the change in log-likelihood. In all cases, the mean of the distribution is within 10%, but the variance of the log-likelihood is quite different from what we would expect for a χ^2 distribution (twice the mean). For example, figure 3.3a shows the case when $K = 4$

in the previous test. In fact, the variance also seems to depend on the community structure of the network, something we haven't taken into account. However, the shape of the distribution does appear to be very similar to that of a squished χ^2 distribution (which is approximately Gaussian for the large parameters we're dealing with) as tested by Q-Q plots, for example as shown in figure 3.3b.

The change in log-likelihood is clearly dependent on the network structure. For example, we compare the difference in log-likelihoods of the previous network with a comparable test on a set of networks with slightly different parameters. In this case, we keep the same number of vertices and average degree, but change the relative group sizes so that 1 of the 3 communities is twice as big as the other two. We also make the communities more assortative: 95% of a vertex's connections on average are now within the same community, and 2.5% of the connections go to each of the remaining two groups. Figure 3.4 compares the two tests. Notice that the change in log-likelihoods is different between the two tests at the same K values. Since the correct communities are easily found in both cases and the networks are mostly comparable, this difference must be due to the different community structure.

As mentioned earlier, AIC can be used to correctly predict the number of communities in the synthetic networks. However, the change in log-likelihood for the first value of K that should not be accepted as correct is significantly different from the amount required for AIC to reject it (in other words it is not the case that $2\Delta\mathcal{L} \approx 2n$). We demonstrate on a real network that this inaccuracy causes AIC to underrepresent the number of communities, whereas our counting heuristic correctly predicts the number of groups.

The network we study is the well-known college football network studied by Girvan and Newman [58]. Vertices in this network are NCAA Division I-A college football teams and an edge is placed between two teams if they played each other during the 2000 football season. Teams in college football are placed into conferences for purposes



(a)

(b)

Figure 3.3: (a) Histogram of the computed change in log-likelihood when $K = 4$ for the test described in the text. The red curve is the best fit to a χ^2 distribution normalized to match the histogram, and is obviously too wide. (b) Corresponding Q-Q plot. The green line is a fit to the quantiles, and the blue line is the line $y = x$. The fit is good, meaning the shape of the distribution is approximately correct, but the blue and green lines don't match up, which means that the width of the distribution is incorrect.

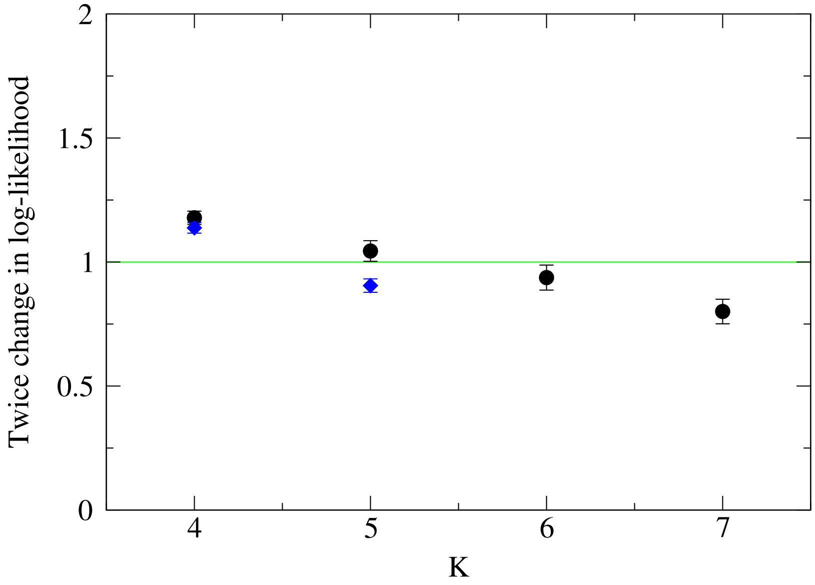
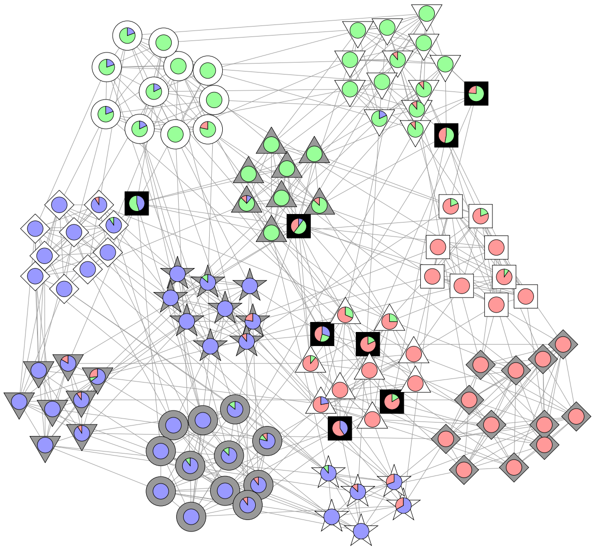


Figure 3.4: Comparison of the two tests mentioned in the text. As in figure 3.2, black squares are the results of the first test, with equal-sized communities, and the green line is the stopping point. Blue diamonds are the results of the test with differently-sized communities. The two tests do not have a strong overlap, showing that a network's community structure impacts the change in log-likelihood. $K = 1, 2$ are not shown since they actually exhibit community structure – we have no reason to believe that the two tests should give the same change in log-likelihood there anyways.

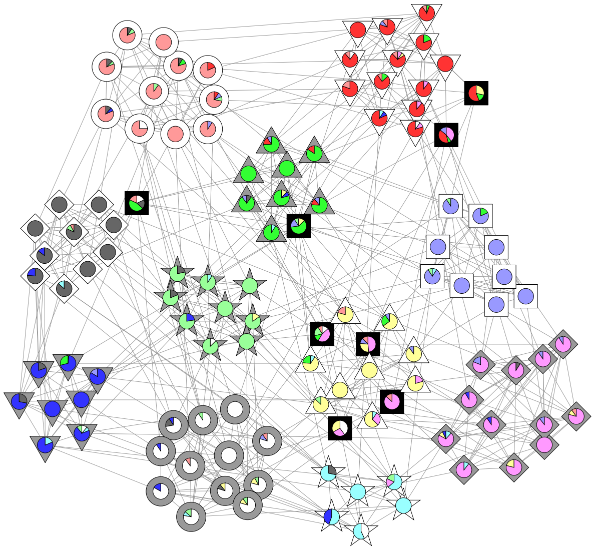
of competition, and it has been repeatedly found that community detection methods can detect these conferences. In 2000 there were eleven conferences. The exception to this structure are the so-called independent teams, teams with no conference affiliation that typically compete against teams from multiple conferences. That year there were eight independent teams.

In Fig. 3.5 we show what happens when you use either AIC (a) or our heuristic with a p-value of 0.5 (b) to extract communities from the college football network. The value of 0.5 is chosen so that it is not affected by inaccuracies in the variance, since the χ^2 distribution is approximately symmetric about its mean, and we will soon discuss this particular choice of p-value and its impact on the fitted structure in further detail. The communities found are plotted as colored vertices (pie charts for vertices in multiple groups, where the fraction of each color is the fraction of edges in that community), and an underlayed shape and color combination is used to display the actual conferences. In particular, black squares are independent teams.

AIC picks off a very wrong solution, fitting only three groups. However that does not mean that the community structure the overlapping community detection blockmodel found isn't especially interesting. The groups correspond to the geographic region of each college, and the conferences are subgroups of the three communities since they are also geographic in nature. The green vertices are teams associated with colleges located in the Midwest and Northeast regions in the U.S., the red vertices are in the South and on the Atlantic coast, and the blue vertices comprise of the colleges on the Pacific coast and Plains states. Student members of the teams must be enrolled in classes to participate with the team, so the distance a team can travel for a game is somewhat limited, so it makes sense that the network should be somewhat geographically clustered. In particular, someone with a knowledge of college football could make the argument that there are three geographic regions that teams play in, and in that regard AIC has found the correct solution.



(a)



(b)

Figure 3.5: Community structure as extracted using the overlapping community detection blockmodel with AIC (a) and our counting heuristic with a p-value of 0.5 (b). Colored circles correspond to the groups, and the actual conferences are shown by the underlayed grayscale shapes. Extracted communities are correlated with the conference structure in both cases, but AIC doesn't find the correct number of conferences whereas our heuristic does. See figure 2.6 for a clear version of the actual conferences. Of special interest are the independent teams, drawn as black squares.

On the other hand, our heuristic finds the correct number of communities as well as the correct communities themselves – depending on your p-value threshold. The last group to be added are the light blue vertices, splitting off from the yellow vertex group with a p-value of $0.1 \sim 0.2$ depending on the exact groupings in the EM algorithm[3]. If the p-value is too relaxed, a 12th group is discovered at about $p = 0.6 \sim 0.7$. Had our heuristic predicted the variance of the distribution of the change in log-likelihood, these two groups would be spread further from 0.5, and we would have better distinguished the number of groups and we would be able to choose p-values more liberally – ideally with a value of $0.05 \sim 0.1$ as is commonly chosen. Additionally, the overlapping community detection model finds interesting behavior in the community structure beyond the well-defined conferences. Several vertices in figure 3.5b appear to belong to the wrong group and/or a large number of groups. Each of these vertices is underlayed with a black square, meaning that they are the independent teams.

3.6 Discussion

In this chapter, we have described a heuristic method for choosing the number of communities in a statistical network model for overlapping community detection. The method works very well, correctly identifying the number of communities in both synthetic and real networks. Since the overlapping community detection model can be applied efficiently to extract communities in a network, this makes it possible to do community detection with an unknown number of communities on large networks. It is possible to make this process even more efficient by applying a community bisection technique or other similar method that takes advantage of the already discovered community structure rather than running the full EM algorithm each time a new

[3] Since this network is relatively small, a change of a single vertex's communities is enough to give it this difference.

community needs to be added.

The heuristic has several drawbacks however. It is not theoretically proven to be the correct answer for this model, and has an unexplained multiplicative factor of two to match our intuition on statistical model techniques to observations of synthetic networks. Furthermore, the variance of the change in log-likelihood distribution isn't accounted for by our heuristic.

There is also the issue that the heuristic relies on the EM algorithm to have found the absolute maximum of the likelihood in both the $K-1$ and K cases. Most of time the p-values are incredibly low so this doesn't cause much concern, but when you're within a couple of communities of the correct answer, you have to start worrying about the total accuracy of not only the communities themselves but also the exact number of communities. This is a general problem with statistical model selection though and does not have an easy solution.

CHAPTER IV

APS multi-modal network

4.1 Introduction

In this chapter, we temporarily step away from the question of blockmodels and instead explore a large multi-modal network like the one shown in Figure 1.4d. It is common to project networks like this into single mode networks where there is only one kind of vertex and edge, and here we demonstrate that you can learn a lot about network dynamics specifically by analyzing across the different types of vertices and edges in addition to looking at them individually. Multi-modal networks are not commonly studied from any perspective, and we hope to garner interest in studying their structure in more depth [140].

Citation networks [120] and coauthorship networks [60, 61, 101] are distinct network representations of bodies of academic literature that have both been the subject of quantitative analysis in recent years. In a citation network the network nodes are papers and a directed edge runs from paper A to paper B if A cites B in its bibliography. In a coauthorship network the nodes are authors and an undirected edge connects two authors if they have written a paper together. Both kinds of network can shed light on habits and patterns of academic research. Citation networks, for instance, can give a picture of the topical connections between papers, while coauthorship networks can shed light on patterns of collaboration such as the size of collaborative groups or the frequency of repeated collaboration.

In this chapter we analyze networks of citation and coauthorship derived from a large data set made available by the American Physical Society (APS), which consists of bibliographic and citation data for the Physical Review family of physics journals and spans the entire history of those journals, more than a hundred years, from their inception in 1893 to 2009 [1]. The data set is unusual both because of the length of time it spans and also because it contains information on both citation and coauthorship for the same body of literature. A number of previous analyses of the data have been published [27, 62, 125] but our work adopts a somewhat different viewpoint from other studies in focusing on the interactions between authorship and citation, as well as on long time-scale patterns in the data. In particular, the simultaneous availability of citation and coauthorship data allow us to associate citations not only with papers but with individual authors, so that we can tell whether or not a particular author cites another. Combining this insight with the temporal aspects of the data we find, for example, that researchers cite their own or coauthors' papers more quickly after publication than they do the work of others; that authors show a strong tendency to return the favor of a citation from another author, especially a previous coauthor; that, contrary to some recent conjectures, having a common coauthor does not make two authors likely to collaborate in future [34, 74, 99]; and that there has not (at least within the journals we study) been any increase over time in self-citations, the number holding roughly constant at about 20% of all citations for over a century.

4.2 The data set

In its raw form the data set we study contains records for 462 090 papers published in the various Physical Review journals, each identified with a unique numerical label. Data for each paper include paper title, date of publication, the published names and

[1] More details about the data set can be found on the web at https://publish.aps.org/datasets

affiliations of each of the authors, and a list of the numerical labels of previous Physical Review papers cited. The data set is unusual in two respects: the long period of time it covers, which spans 116 years from 1893 to 2009, and the fact that it includes citation data and hence allows us to compare coauthorship patterns with citations, at least for that portion of the citation network that appears in the Physical Review—citations to and from non-Physical-Review journals, of which there are many, are not included.

Before performing any analysis, however, there are some hurdles to overcome. Foremost among them is the fact that the name of an author alone does not necessarily identify him or her uniquely. Two authors may have the same name, or the same author may be identified differently in different publications (with or without a middle initial, for example). Unlike some journals, such as those of the American Mathematical Society[2], the Physical Review does not maintain unique author identifiers that can be used to attribute authorship unambiguously. As a first step in analyzing the data, therefore, we have processed it using a number of disambiguation techniques in order to infer actual author identity from author names as accurately as possible. Details of the disambiguation process are given in Appendix E.

In addition, we have performed a modest culling of the data to remove outliers, the most substantial action being the removal of all papers with fifty or more authors, which are primarily recent papers in experimental high-energy physics. (Almost all of them, about 91%, were published either in Physical Review D, which covers high-energy physics, or Physical Review Letters; the remainder were in Physical Review C, which covers nuclear physics.) As we show shortly, though papers with more than fifty authors are only a small fraction of the whole (about 0.7%), their inclusion skews results for the last thirty years substantially by comparison with the rest of the time period. For results whose outcome depends strongly on the presence or not of these

[2] A description of the unique author identifier system used by the American Mathematical Society can be found at http://www.ams.org/publications/math-reviews/mr-authors.

	All papers	Papers with 50 authors or fewer
Total papers	460889	457516
Total authors	235533	226641
Authors per paper	5.35	3.34
Citations per paper	10.16	10.16
Number of collaborators	59.44	17.24
Papers per author	10.47	6.74

Table 4.1: Mean values of some statistics for our data set, with and without papers having over 50 authors.

papers, we quote results both with and without, for comparison.

Table 4.1 gives some basic parameters of the resulting data set.

4.3 Analysis

In the next few sections we present a variety of analyses of the Physical Review data set. We begin by looking at some basic parameters of authorship and coauthorship.

4.3.1 Authorship patterns

Figure 4.1 shows a cumulative distribution function for the number of papers an author publishes, aggregated over the entire data set. That is, the figure shows the fraction of authors who published n papers or more as a function of n, which is a crude measure of scientific productivity. The axes in the figure are logarithmic, and the approximate straight-line form of the distribution function implies that scientific productivity follows, roughly speaking, a power law, a result known as Lotka's law, first observed by Alfred Lotka in 1926 [91] and confirmed by numerous others since. (It has also been suggested that the distribution is log-normal rather than power-law [130]. It is known to be hard to distinguish empirically between log-normal and power-law distributions [30].) In Figure 4.1 we give separate curves with and without

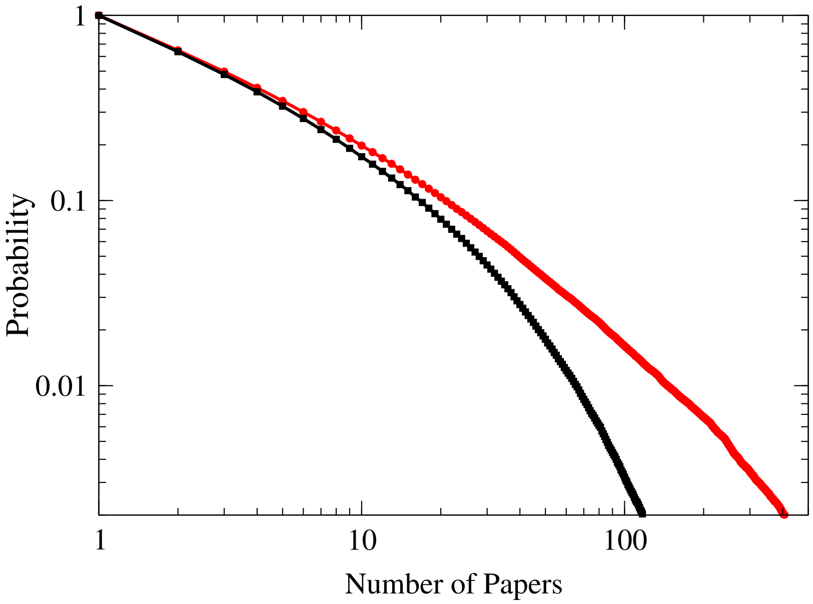


Figure 4.1: Probability that an author wrote more than a given number of papers. Red circles indicate values calculated from the full data set; black squares are values from the data set after papers with fifty or more authors have been removed. The plot is cut off around 500 papers because there is very little data beyond this point.

the papers that have fifty or more coauthors. As the figure shows, the difference between the two is primarily in the tail of the distribution, among the authors who have published the largest number of papers, indicating that a significant fraction of the most productive authors are those in large collaborations. In fact, if one compiles a list of the fifty authors publishing the largest numbers of papers, only one of them remains on that list after papers with fifty or more authors are excluded. This probably results from a combination of two effects: first, larger groups can publish more papers simply because they have more people available to write them; and second, a large and productive group of collaborators contributes many apparently prolific authors to the statistics—each of the many coauthors separately gets credit for being highly productive. It is precisely because of biases of this kind that we exclude papers with many authors from some of our calculations.

We can remedy this problem to some extent by measuring productivity in a more sophisticated fashion. Rather than just counting up all the papers an author was listed on, we can instead divide up the authorship credit for a paper among the contributing authors so that, for example, each author on a two-author paper is credited with half an authorship for that paper. This reduces significantly the impact of large collaborations on the statistics, though the distribution of number of papers authored is still highly skewed, with certain authors producing much more science than others. A common way to visualize such skewed distributions is to use a Lorenz curve, a plot of the fraction of papers produced by the most prolific authors against the fraction of authors that produced them. Such a curve is shown for our data set in Figure 4.2, and the sharp rise in the curve at the left-hand side indicates the concentration of scientific productivity among the most productive scientists. Note for instance that productivity appears roughly to follow the so-called 80–20 rule, such that about 80% of the output is produced by the 20% most productive authors. Notice also that there is almost no difference in the Lorenz curves with and without the 50-plus-author papers, precisely because we have divided up the authorship credit so that the effect of many-author papers is diminished.

The distribution can be further quantified by measuring a Gini coefficient, which is defined as the excess area under the Lorenz curve compared to the case where everyone has the exact same productivity. In our data set, the Gini coefficient is 0.70, a relatively large figure as such coefficients go, indicating high skew. (Gini coefficients for wealth inequality, for example, which is the context in which such coefficients are perhaps best known, rarely rise above 0.6, even in the most inequitable countries.)

The data set also allows us to measure the productivity of the entire field of physics over time, something that cannot be done with many other data sets. Figure 4.3 shows the total number of papers published in the Physical Review in five year time blocks since 1893. With the important caveat that these results are for a single collection of journals only, and one moreover whose role within the field has evolved during

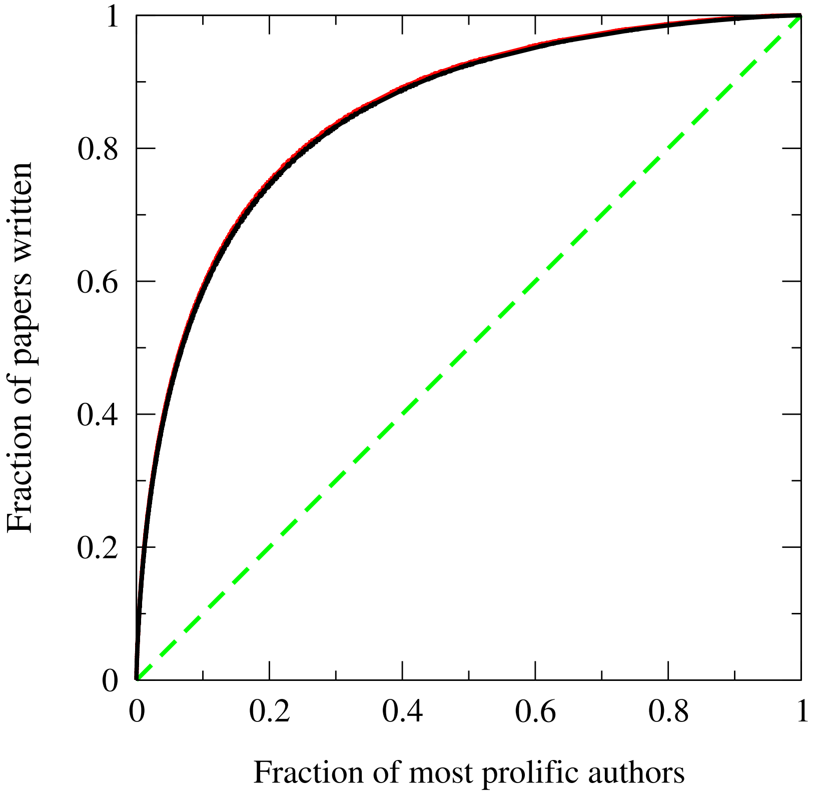


Figure 4.2: Fraction of papers written by the most prolific authors (with credit for multi-author papers divided among coauthors, as described in the text). The red (grey) curve represents values calculated from the full data set; the black curve represents values after papers with fifty or more authors have been removed. Note that the two curves are almost indistinguishable. The dashed line indicates the form the curve would take if all authors published the same number of papers.

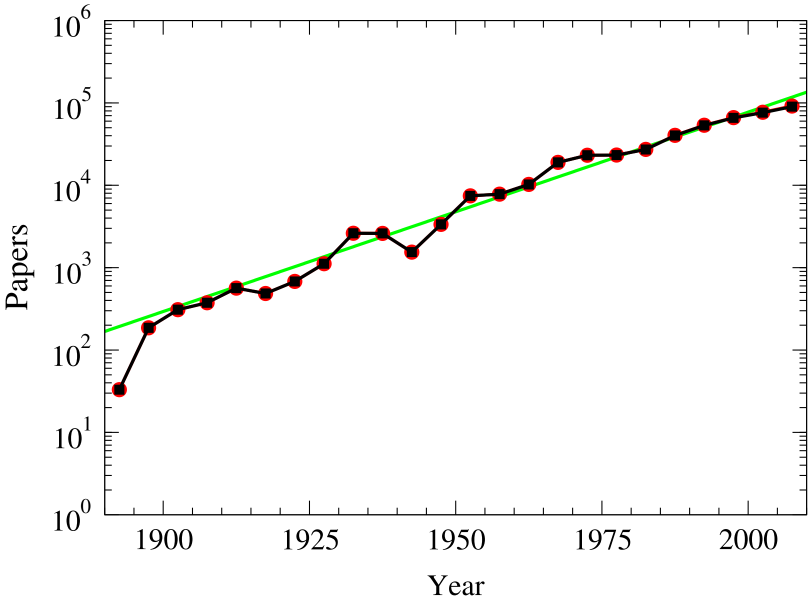


Figure 4.3: Number of papers published in each five-year block. Red circles indicate numbers calculated from the full data set; black squares are calculated from the data set after papers with fifty or more authors have been removed. Note that the two values are almost indistinguishable. The straight line is the best-fit exponential.

its history from provincial up-start to one of the leading physics publications on the planet, we see that there is a steady increase in the volume of published work, which appears roughly to follow an exponential law (a straight line on the semi-logarithmic scales of the figure). An interesting feature is the dip in the curve in the 1940s, which coincides with the second World War, followed by a recovery in the 1950s, perhaps attributable in part to increased science funding in the postwar period. The combined result of these deviations, however, is only to put the curve back on the same path of exponential growth after the war that it was already on before it. In his early studies of secular trends in scientific output, Derek de Solla Price [118, 119] noted a similar exponential growth interrupted by the war, and measured the doubling time of the growth process to be in the range from 10 to 15 years. The best exponential fit to our data gives a compatible figure of 11.8 years.

Figure 4.4 shows the corresponding plot of the number of unique authors in the

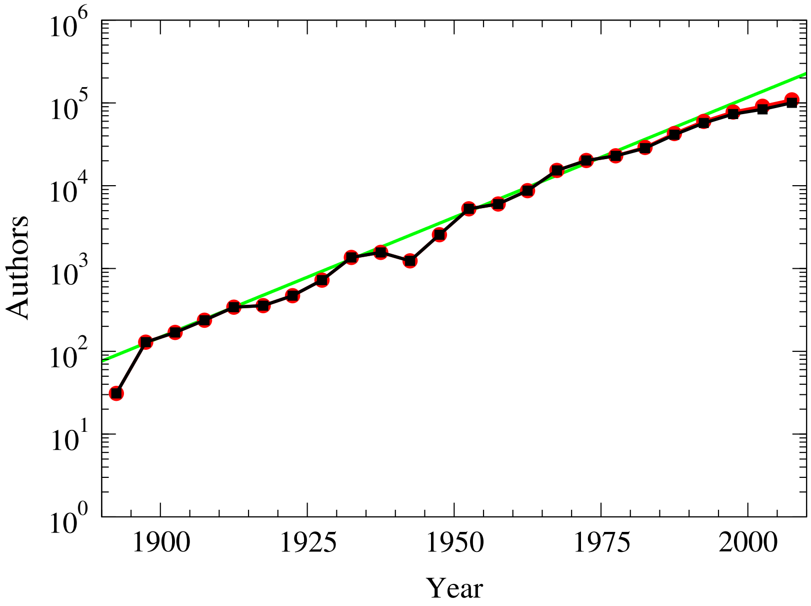


Figure 4.4: Number of unique authors who published a paper in each five-year block. Red circles indicate numbers calculated from the full data set, while black squares are calculated from the data set after papers with fifty or more authors have been removed. Note that the two values are almost indistinguishable. The straight line is the best-fit exponential.

data set in each five-year block as a function of time. Like the number of papers published, the number of authors appears to be increasing exponentially, and with a roughly similar (but slightly smaller) doubling time of 10.4 years. Thus, despite the marked increase in productivity of the field as a whole, it appears that each individual scientist has produced a roughly constant, or even slightly decreasing, number of papers per year over time.

The natural complement to measurement of the number of papers per author is measurement of the number of authors per paper, i.e., the size of collaborative groups. Figure 4.5 shows the mean number of authors per paper in our data set as a function of time, and there is a clear increasing trend throughout most of the time period covered, with the average size of a collaborative group rising from a little over one a century ago to about four today. A similar effect has been noted previously by, for example, Grossman and Ion [60], for the case of mathematics collaborations. In

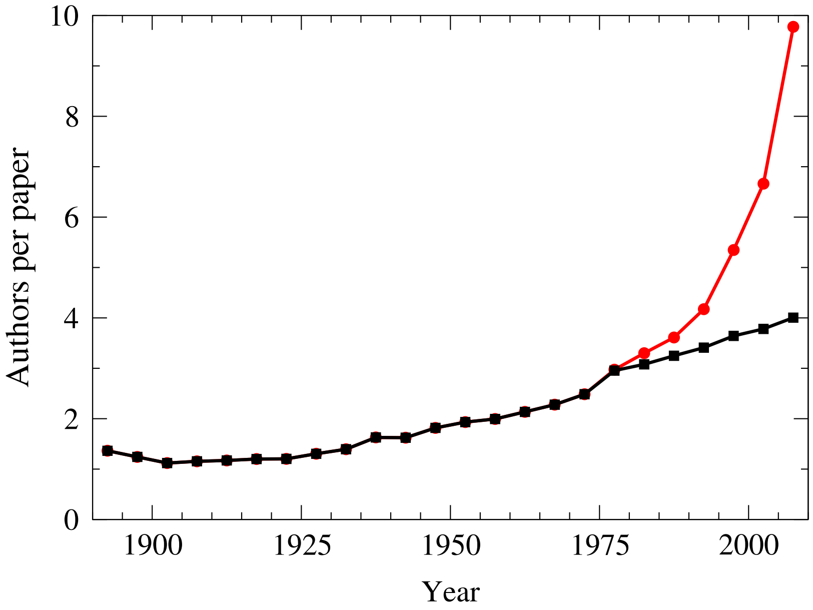


Figure 4.5: Number of authors per paper averaged over five-year blocks. Red circles indicate the full data set; black squares are the data set after papers with fifty or more authors have been removed.

our calculations we have again calculated separate curves with and without papers having fifty or more authors and a comparison between the two reveals a startling effect: while there is almost no difference at all between the curves prior to about 1975, there is a large and rapidly growing gap between them in the years since. Without these papers the growth in group sizes has been slow and steady for decades; with them it departs dramatically from historical trends after the 1970s, indicating a large and growing role in physics (or at least in physics publication) for big collaborations.

An alternative view of the same trend is given in Figure 4.6, which shows the number of unique coauthors an author has, on average, during each five year time block. Every coauthor in a time block is counted, even if he or she was also counted in a previous time block (but previous coauthors are not counted unless they are also coauthors in the new time block). As the figure shows, this number has also risen significantly over the last century, from a little over one to more than ten today (and more than sixty if one includes collaborations with fifty or more members). Since we

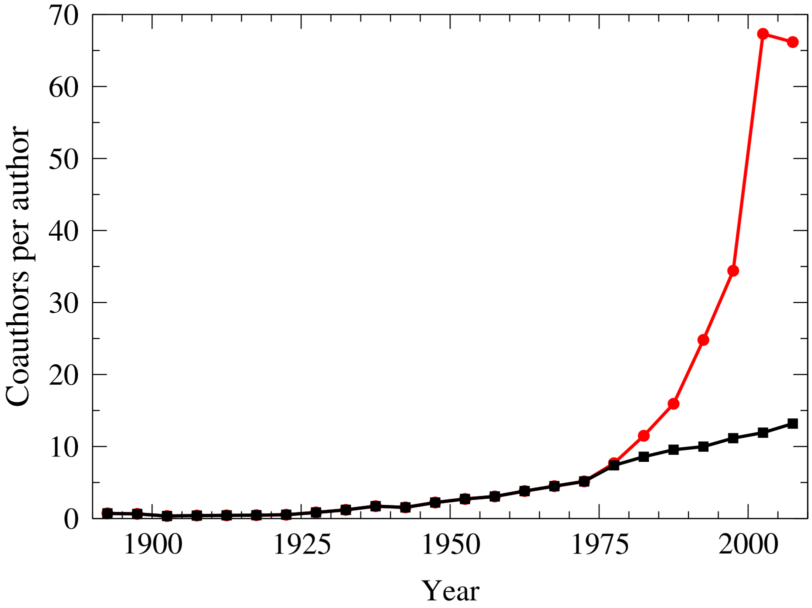


Figure 4.6: Average number of unique coauthors of an author, averaged in five-year blocks. Red circles indicate the full data set; black squares are the data set after papers with fifty or more authors have been removed.

only have data from the Physical Review, it is likely that we miss some collaborators, so these numbers are in practice only lower bounds on the actual numbers.

4.3.2 Citation patterns

Let us now add the citation portion of the data set to our analyses and examine citation patterns over time in the Physical Review, as well as interactions between citation and coauthorship.

Figure 4.7 shows the average number of citations by a paper and to a paper, over the time period covered by the Physical Review data set. The black curve, the number of citations that a paper makes, shows a steady increase over time—authors used to cite fewer papers and have been citing steadily more in recent decades. One possible explanation for this phenomenon is the increase in the volume of literature available to be cited, although it has also been conjectured that authors have been under greater pressure in recent decades, for example from journal editors or referees,

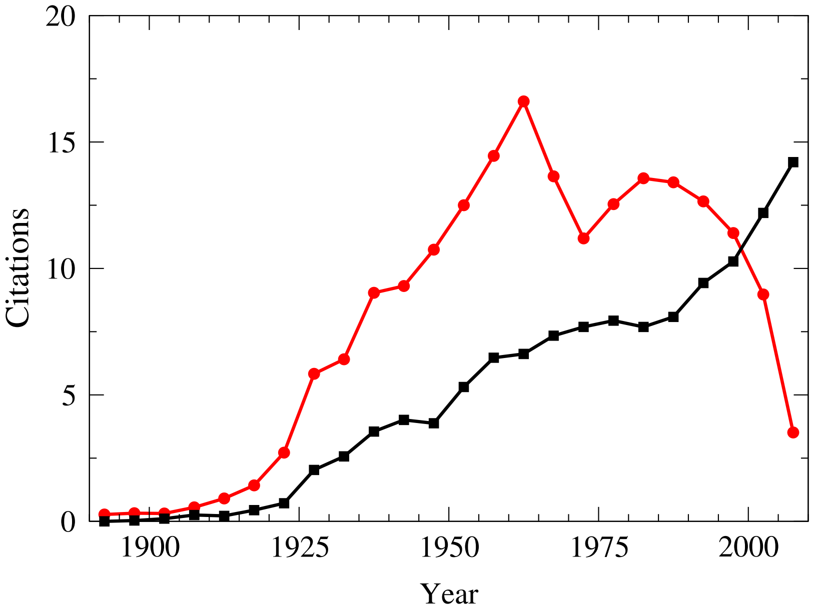


Figure 4.7: Average numbers of citations made (black squares) and received (red circles) per paper, in five-year blocks.

to add more copious citations to papers [148].

The red curve in Figure 4.7 is the average number of citations received by a paper, which shows more irregular behavior, rising to a peak twice before dropping off in recent times. A number of effects are at work here. First, if (as we will shortly see) most citations are to papers in the recent past, then a steady increase in citations *by* papers should lead to an increase in citations *to* papers published slightly earlier. Behavior of this kind has been observed in previous studies, such as the comprehensive study by Wallace *et al.* using data from the Web of Science [139]. The growth in number of citations received cannot continue to the very end of the data set, however, since the most recent papers are too recent to have accrued a significant number of citations and hence we expect a drop at the rightmost end of the curve, as seen in the figure.

There is, however, also a notable dip in the red curve around 1970, whose origin is less clear. (It is not seen, for instance, in the work of Wallace *et al.*) In examining the data for this period in detail, we find that the dip in citations per paper is due primarily to an increase in the number of papers published in the Physical Review (which expanded considerably during this period), while the number of citations received by those papers, in aggregate, remains roughly constant. The increase in papers published may have been in part a response to the general expansion of US physics research during the 1960s, following the establishment of the National Science Foundation, but the data indicate that the greater volume of research did not, at least initially, result in a greater number of citations received, and hence the ratio of the two displays the dip visible in Figure 4.7. However, the upward trend in the curve reestablishes itself from about 1970 onward, suggesting that in the long run there was an increase not only in the number of papers published, but also in the number that are influential enough to be later cited.

It is interesting to compare the data for citations received with the predictions of theoretical models for the citation process. Perhaps the best known class of models are the preferential attachment models [13], and particularly the 1976 model of Price [117], a simple model in which the rate at which a paper receives citations is assumed to vary linearly with the number it already has. In its most naive application, this model makes predictions that differ strongly from the observations plotted in Figure 4.7. The model predicts that the largest number of citations should go to the oldest papers and the smallest to the youngest, so that the red curve in the figure should be monotonically decreasing. There are a number of possible explanations for the disagreement. A popular theory is that papers "age" over time, becoming less well cited as they become older [128, 156], perhaps because their field has moved on to other things, because they have been superseded by more advanced or accurate work, or because their results are so well known that authors no longer feel the need to cite them. Were this the case, most citations would be to recent papers, and the curve of citations received would mostly mirror the curve of citations given, albeit with a time lag whose length would be set by the rate at which papers age. An alternative theory, for which there is some empirical evidence, is that preferential attachment models do represent citation patterns quite well within individual subfields [104], but not when applied to the literature as a whole. A central parameter in the preferential attachment models is the date of the start of a subfield, and since different subfields have different start dates, the model might be expected to work within subfields but not for the overall data set.

Figure 4.8 tests the aging of papers within the Physical Review data set by plotting the fraction of citations that are to papers a certain time in the past. Let us focus for the moment on the black curve, which includes all citations in the entire data set. The figure shows that there does indeed appear to be a strong aging effect, with the citation rate dropping off approximately exponentially over time (which would be a straight line on the semi-logarithmic scales of the plot). This finding is in agreement with previous studies of aging [156], which also found exponential decay. An alternative interpretation of the data, however, is that there is no aging occurring at all, and that the drop in citations is a purely mechanical effect that results from dilution of the literature—in a small, young field there are only a few papers to cite and hence each receives a lot of citations; in an older field there are more papers and so individual citation rates fall off. To the extent that it has been tested, the latter theory appears to agree well with available citation data and also with the prediction of the preferential attachment models [82], so at present the evidence for (or against) aging in our data set is inconclusive.

4.3.3 Interactions between citation and coauthorship

Perhaps the most interesting aspect of the Physical Review data, however, is the window it gives us on the interplay between citation and coauthorship. One way to probe this interplay is to divide citations according to the collaborative roles assumed by the authors of the citing and cited papers and then compare the resulting cita-

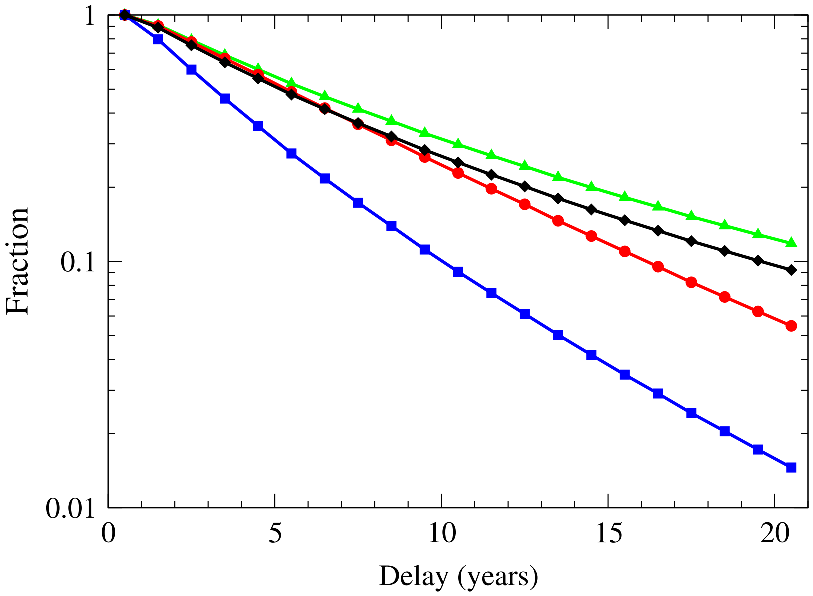


Figure 4.8: Fraction of citations made more than a given number of years after publication. Black diamonds include all citations, blue squares are self-citations, red circles are co-author citations, and green triangles are distant citations.

tion patterns. In the present work, we divide citations into three classes, following Wallace *et al.* [140]: self-citations, where the citing and cited papers shared at least one coauthor; coauthor citations, where at least one author of the citing paper has previously collaborated with at least one author of the cited paper (but there are no common authors between papers, so that self-citations and coauthor citations are disjoint); and distant citations, which includes all citations other than self-citations and coauthor citations. (Other authors who have examined citation and collaboration have gone further and considered also citations between coauthors of coauthors [140], but this proves computationally unfeasible in the present case because of the size of the Physical Review data set.) We emphasize that we only consider individuals to be coauthors if they have *previously* coauthored when the citation occurs. Coauthorship that comes after the citation is not counted. Also our data are limited to the Physical Review, so the number of coauthor citations will in reality be higher than presented here, both because some citations are missing from our data and because some coauthorships are.

Figure 4.9 shows the fraction of citations that fall into each of the three classes as a function of the year of publication of the citing paper. Roughly speaking, the three curves appear flat over time. There is a modest increase in the fraction of coauthor citations (the lowest, red curve in the figure), but this can be explained by the increase in the number of coauthors available for citation, shown in Figure 4.6, which is of a similar magnitude. In other respects, the rule of thumb seems to be that a constant 20% or so are self-citations, 75 or 80% are distant citations, and the small remaining fraction are to coauthors.

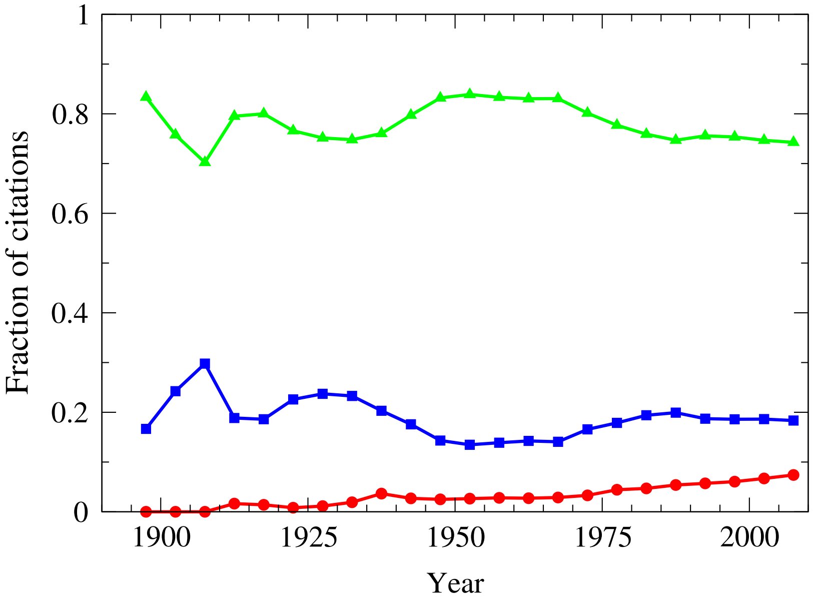


Figure 4.9: Fraction of citations made, by type, in five-year blocks. There were no citations made in the 1890–1894 block. Blue squares represent self-citations, red circles are co-author citations, and green triangles are distant citations.

The distribution of time between the publication dates of a new paper and the papers it cites is shown for the three classes of citation in Figure 4.8, as the blue, red, and green curves. Here we do notice a significant difference between the classes. In particular, the self-citations (in blue) fall off faster than coauthor and distant citations. This implies that a larger fraction of self-citations occur rapidly after publication, compared with citations in the other classes. This is not unexpected, given that a researcher presumably knows about their own research sooner, and in more detail, than they know about others'. We note also that coauthor citations are slightly earlier than distant citations, which again seems reasonable. One must be careful in the interpretation of these results, however. An alternative explanation for the same observations is that a paper can be cited by others long after the author retires or leaves the field, which could make the average delay for citations by others longer than that for self-citation. There is no way to tell, purely from the delay statistics themselves, which explanation is the better one.

Table 4.2 summarizes the mean delay to citation for the three citations classes. We explore the differences between citation classes further in the next section.

Citation type	Mean delay (years)
Self-citations	4.12
Coauthor citations	6.92
Distant citations	9.02
All citations	7.89

Table 4.2: Mean time delay between a paper's publication date and the dates of the papers it cites.

4.3.4 Self-citation and coauthor citation

Consider Table 4.3, which gives the percentages of papers that make or receive at least one self-citation or coauthor citation, provided that such a citation is possible. Nearly 70% of papers cite at least one paper by the same author (or one of the same authors, if there are several), and 60% of them receive such a citation. These numbers may at first appear large, and raise concerns, given the use of citation counts as a measure of impact, that authors might be inflating their counts by self-citing [15, 67]. But taken with the fact that the number of citations per paper and the fraction which are self-citations are both sizable, these large numbers are not unexpected. Figure 4.9 shows that overall self-citation has remained constant and moderate, around 20%, and that there has been no sizable recent excess in self-citation.

Citation type	Made (%)	Received (%)
Self-citation	68.9	60.3
Coauthor citation	42.0	31.3
Both	35.6	26.3
Either	75.0	64.2
Either given both possible	76.4	66.4

Table 4.3: Percentage of papers that make or receive at least one citation of a given type.

A more interesting question is whether researchers have a tendency to reciprocate citations by others. If author A cites a paper of author B, does B return the favor by later citing A? To address this question we measure the fraction of citations of one author by another (excluding citations of one's own papers) that are reciprocated in one or more later publications. We calculate separate figures for pairs of authors who have previously co-authored a paper and those who have not and find that 13.5% of citations between non-coauthors are reciprocated when possible, while an impressive 43.8% of citations between coauthors are reciprocated. (Keep in mind that no authors can overlap between a citing and a cited paper for the citation to be considered a coauthor citation and not a self-citation.) Both these numbers are very high compared to the expected reciprocity if citations were made uniformly at random, but this doesn't necessarily imply a tit-for-tat return of citations. A citation is presumptively an indication that two papers fall in similar subject areas, and thus the presence of a citation greatly increases the chances that the authors are working in the same area, which in turn increases the likelihood of citation in general and therefore the chances of reciprocated citation. In the case of previous coauthors the chances of working in the same field are likely even higher. Unfortunately, we currently do not have any model of the citation process detailed enough to make a quantitative prediction of the size of this effect against which we could compare our measurements to test for significance.

4.3.5 Transitivity

Transitivity, in the context of networks, refers to the observation that "the friend of my friend is also my friend" [143]. In the context of coauthorship, for example, it is observed that if A has coauthored a paper with B and B with C, then A and C are more likely also to have coauthored a paper. One can define a so-called clustering coefficient that quantifies this effect, measuring the average probability that the friend of your friend is also your friend [146], and such coefficients have been measured in many networks [12, 34, 80, 110]. Typically one finds that the values are significantly higher than one would expect if network connections were made purely at random, and our coauthorship network is no exception. For the data set studied here we find a clustering coefficient of 0.212, which is comparable with other figures reported for coauthorship networks [101].

In this case, however, the nature of the data set allows us to go further. The conventional explanation for high transitivity in networks relies on a triadic closure mechanism, under which two authors who share a common coauthor are more likely to collaborate in future, perhaps because they revolve in the same circles, attend the same conferences, work at the same institution, or are introduced to one another by their common acquaintance [34, 74, 99]. The present data set's time-resolved nature allows us to test this hypothesis directly. We can calculate what fraction of the time individuals who share a common coauthor but have not previously collaborated themselves later write a paper together. When we make this measurement for the Physical Review data we find the fraction of such author pairs to be only 0.0345—a much smaller fraction than the clustering coefficient of the whole network reported above. One reason for this small figure is that a large fraction of the transitivity seen in coauthorship networks comes from papers with three or more authors, which automatically contribute closed triads of nodes to the coauthorship network. Such triads however are excluded from our calculation of the probability of later collaboration. The large difference between the two probabilities we calculate implies that only a small fraction of the network transitivity comes from true triadic closure processes.

Nonetheless, the triadic closure process does appear to be present in our data set. Figure 4.10 shows the probability of future coauthorship between two individuals as a function of their number of common coauthors, and we see that the probability increases sharply, a finding that is consistent with previous results [20, 99].

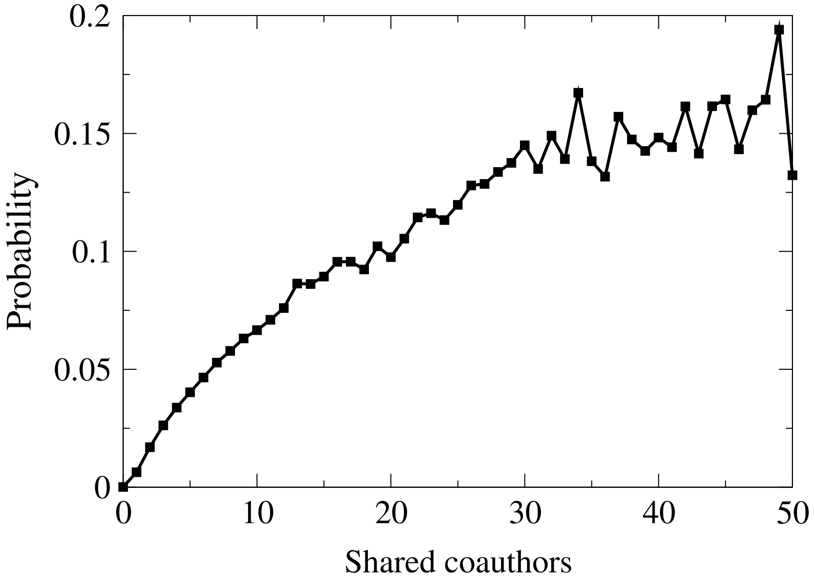


Figure 4.10: Probability of future coauthorship with another author as a function of the number of shared coauthors. The number of shared coauthors is counted at the time of first coauthorship or the date of either coauthor's last published paper, whichever comes first.

4.4 Discussion

In this chapter we have analyzed a large data set from the Physical Review family of journals, taking a network perspective. Rather than focus solely on either citation or coauthorship networks, as most previous studies have done, we have instead combined the two, which allows us to study questions about the ways in which people—and not just papers—cite one another, and the extent to which scientists collaborate with those they cite or cite those with whom they collaborate. The time-span of the data set is unusually large, covering more than a century of publication, which allows us to study long-term changes in collaboration and citation patterns that are not accessible with smaller data sets.

Our main findings are that the Physical Review appears to be growing exponentially, with a doubling rate slightly less than 12 years, and the number of citations per paper within the journals also appears to be growing. The fraction of self-citations and citations among coauthors is more or less constant over time, and authors tend to cite their own papers sooner after publication than do their coauthors, who in turn cite sooner than non-coauthors. We observe a strong tendency towards reciprocal citations, researchers who cite another author often receiving a citation in return later on, with especially high rates for citations between coauthors. Contrary to some previous claims [34, 74, 99], however, there is only a small triadic closure effect in the coauthorship patterns; two researchers who share a common coauthor but have never collaborated themselves have only a rather small probability of collaborating in future—about 3.5%. This number is nonetheless much higher than the probability for two randomly chosen researchers, and moreover increases sharply as the number of common coauthors increases.

A limitation of our analysis is that the data we use come from a single family of journals in a single field. There are, however, some results for other journals and fields that suggest that the patterns we observe extend beyond physics and the Physical Review. In one recent study, for example, Huang *et al.* [77] examined a collection of papers in computer science drawn from the CiteSeer database of online preprints. They find, as we also do, that the number of papers and number of authors both increase roughly exponentially over time, while the number of authors per paper and number of coauthors per author increase roughly linearly. Wuchty *et al.* [150] examined a large set of papers drawn broadly from the sciences and engineering, using data from the commercial Web of Science database (formerly the Science Citation Index). They observe in particular that the average number of authors on a paper has increased steadily over time, at least for papers with more than one author, which again agrees qualitatively with our observations. Döbler [138] studied a data set representing the fields of mathematics, logic, and physics from 1800 to 1998 and found again that collaboration has increased over time, albeit intermittently, and at a rate that depends on the field.

There are many other questions that could be addressed with the data we have analyzed, the unusually long time-span and combination of publication and citation data opening up a variety of possibilities. For instance, we know which papers are published in which of the various Physical Review journals, and hence we have a crude measure of paper topic, which would allow us to answer questions about how the patterns of coauthorship and citation vary between fields within physics. We could also study geographical variations by making use of the data on authors' institutional affiliations [113]. Our analysis of long-term historical trends could also be extended; for the researcher interested in the history of US physics, there are, no doubt, many interesting signatures of historical events hidden within the data. The data set also offers the possibility of tracking the careers of individual scientists, possibly over long periods of time, or of tracking research on a particular topic. And finally, any of our analyses could be extended to data sets that cover other journals or fields other than physics, if and when such data become available. All of these would make excellent subjects for future investigation.

CHAPTER V

Social ranking model

5.1 Introduction

A social network, in the most general sense of the term, consists of a group of people, variously referred to as nodes or actors, connected by social interactions or ties of some kind [143]. In this chapter we consider networks in which the ties represent friendship. Friendship networks have been the subject of scientific study since at least the 1930s. A classic example can be found in the studies by Rapoport and Horvath [124] of friendship among schoolchildren in the city of Ann Arbor, Michigan in the 1950s and 60s, in which the investigators circulated questionnaires among the students in a school asking them to name their friends. Many similar studies have been done since then, with varying degrees of sophistication, but most employ a similar questionnaire-based methodology. A counterintuitive aspect of the resulting networks is that they are directed. Person A states that person B is their friend and hence there is a direction to the ties between individuals. It may also be that person B states that person A is their friend, but it does not have to be the case, and in practice it turns out that a remarkably high fraction of claimed friendships are not reciprocated. In the networks we study in this chapter the fraction of reciprocated ties rarely exceeds 50% and can be as low as 30%.

This could be seen as a problem for the experimenter. One thinks of friendship as a two-way street—a friendship that goes in only one direction is no friendship at all. How then are we to interpret the many unreciprocated connections in these networks? Are the individuals in question friends or are they not? One common approach is simply to disregard the directions altogether and consider two individuals to be friends if they are connected in either direction (or both) [3]. In this chapter, however, we take a different view and consider what we can learn from the unreciprocated connections. It has been conjectured that, rather than being an error or an annoyance, the pattern of connections might reflect underlying features in the structure or dynamics of the community under study [35, 45, 75].

Working with a large collection of friendship networks from US schools, we find that in every network there is a clear ranking of individuals from low to high such that almost all friendships that run in only one direction consist of a lower-ranked individual claiming friendship with a higher-ranked one. We conjecture that these rankings reflect a measure of social status and present a number of results in support of this idea. For instance, we find that a large majority of reciprocated friendships are between individuals of closely similar rank, while a significant fraction of unreciprocated friendships are between very different ranks, an observation consistent with qualitative results in the sociological literature going back several decades [35]. We also investigate correlations between rank and other individual characteristics, finding, for example, that there is a strong positive correlation between rank and age, older students having higher rank on average, and between rank and overall popularity, as measured by total number of friends.

The outline of the chapter is as follows. First, we describe our method of analysis, which uses a maximum-likelihood technique in combination with an EM algorithm to extract rankings from directed network data. Then we apply this method to school friendship networks, revealing a surprisingly universal pattern of connections between individuals in different schools. We also present results showing how rank correlates with other measures. Finally we give our conclusions and discuss possible avenues for future research.

5.2 Inference of rank from network structure

Consider a directed network of friendships between n individuals in which a connection running from person A to person B indicates that A claims B as a friend. Suppose that, while some of the friendships in the network may be reciprocated or bidirectional, a significant fraction are unreciprocated, running in one direction only, and suppose we believe there to be a ranking of the individuals implied by the pattern of the unreciprocated friendships so that most such friendships run from lower to higher rank. One possible way to infer that ranking would be simply to ignore any reciprocated friendships and then construct a minimum violations ranking of the remaining network [6, 126]. That is, we find the ranking of the network nodes that minimizes the number of connections running from higher ranked nodes to lower ranked ones. In practice this approach works quite well: for the networks studied in this chapter the minimum violations rankings have an average of 98% of their unreciprocated friendships running from lower to higher ranks and only 2% running the other way. By contrast, versions of the same networks in which edge directions have been randomized have about 10% of edges running the wrong way on average. (Statistical errors in either case are 1% or less, so these observations are highly unlikely to be the results of chance.)

The minimum violations ranking, however, misses important network features because it focuses only on unreciprocated friendships. In most cases there are a substantial number of reciprocated friendships as well, as many as a half of the total, and they contain significant information about network structure and ranking. For example, as we will see, pairs of individuals who report a reciprocated friendship are almost always of closely similar rank. To make use of this information we need a more flexible and general method for associating rankings with network structure. In this chapter we use a maximum likelihood approach defined as follows.

Mathematically we represent the distinction between reciprocated and unreciprocated friendships in the network using two separate matrices. The symmetric matrix **S** will represent the reciprocated connections—undirected edges in graph theory terms—such that $S_{ij} = S_{ji} = 1$ if there are connections both ways between nodes i and j, and zero otherwise. The asymmetric matrix **T** will represent the unreciprocated (directed) edges with $T_{ij} = 1$ if there is a connection to node i from node j (but not *vice versa*), and zero otherwise. The matrices **S** and **T** are related to the conventional adjacency matrix **A** of the network by $\mathbf{A} = \mathbf{S} + \mathbf{T}$.

Now suppose that there exists some ranking of the individuals, from low to high, which we will represent by giving each individual a unique integer rank in the range 1 to n. We will denote the rank of node i by r_i and the complete set of ranks by R. We have found it to be a good approximation to assume that the probability of friendship between two individuals is a function only of the difference between their ranks. We specifically allow the probability to be different for reciprocated and unreciprocated friendships, which acknowledges the possibility that the two may represent different types of relationships, as conjectured for instance in Davis and Leinhardt [35] and Dijkstra [41]. We define a function $\alpha(r_i - r_j)$ to represent the probability of an undirected edge between i and j and another $\beta(r_i - r_j)$ for a directed edge to i from j. Since $\alpha(r)$ describes undirected edges it must be symmetric $\alpha(-r) = \alpha(r)$, but $\beta(r)$ need not be symmetric.

If we were not given a network but we were given the probability functions α and β and a complete set of rankings on n vertices, then we could use this model to generate—for instance on a computer—a hypothetical but plausible network in which edges appeared with the appropriate probabilities. In effect, we have a random graph model that incorporates rankings. In this chapter, however, we want to perform the reverse operation: given a network we want to deduce the rankings of the nodes and the values of the functions α and β. To put that another way, if we are given a network and we assume that it is generated by our model, what values of the rankings and probability functions are most likely to have generated the network we observe?

This question leads us to a maximum likelihood formulation of our problem, which we treat using an expectation–maximization (EM) approach in which the ranks R are considered hidden variables to be determined and the functions α and β are parameters of the model.

This approach for estimating hidden variables, applying maximum likelihood methods to a simplified generative model, is common within statistics and machine learning, but perhaps less common in other areas, so it may be worthwhile to briefly place our work in context. The model presented here is, clearly, not an accurate representation of the true process of friendship formation. Unquestionably there are other factors besides rank that play into people's decisions to become friends. A realistic model of the process would certainly be more complicated and have many more parameters. Nonetheless, simplified models such as this one are not only common in data analysis, but they also appear to work well in practice. If one is interested in estimating a set of hidden or latent variables, unobserved in the data but nonetheless affecting the observations, then a model incorporating only the effects of those latent variables plus a minimal set of other basic assumptions often turns out to produce useful estimates, and this perhaps surprising observation forms the foundation for a large part of modern statistics. Certainly it is possible to simplify models too far—the standard stochastic block model of community structure in networks is an example of a model that in many cases does not incorporate enough basic features to produce good fits to real-world data for any parameter values and hence fails to extract latent structure even in simple and well-understood examples [83]. In many other cases, however, and particularly in the case of the present chapter, a simplified model that identifies the crucial features and leaves out the rest, can impart substantial insight while avoiding unnecessary elaboration.

The particular model described in this chapter is only one of many that have been proposed to explain patterns of reciprocation in social network data. One of the best known previous models is the so-called p1 model of Holland and Leinhardt [73], an early example of what would now be called an exponential random graph that incorporates parameters governing vertex degrees and a single parameter controlling the probability of reciprocated friendships. A variety of extensions of this model, including some quite elaborate ones, have been subsequently proposed, such as those of Wang and Wong [142] and Strauss and Ikeda [136]. See Wasserman and Pattison [144] for a useful introduction to this area along with a wide range of examples, and Snijders [132] for a recent review. None of these models, however, employs ranking directly as a latent variable, and many of them lean in the opposite direction to our current goal of simplicity in model design, incorporating a range of elaborations that can increase the precision of the fit but also make interpretation more challenging.

With these considerations in mind, let us return to the model proposed in this chapter. We use a Poisson formulation of the model (which gives significantly simpler formulas while being asymptotically identical to a Bernoulli formulation for large networks) in which the likelihood of generating a network G with rankings R, given the functions α and β, is

$$P(G,R|\alpha,\beta) = \prod_{i>j} \frac{[\alpha(r_i - r_j)]^{S_{ij}}}{S_{ij}!}\,\mathrm{e}^{-\alpha(r_i-r_j)} \prod_{i\neq j} \frac{[\beta(r_i - r_j)]^{T_{ij}}}{T_{ij}!}\,\mathrm{e}^{-\beta(r_i-r_j)}. \tag{5.1}$$

Note that we have excluded self-edges, since individuals cannot name themselves as friends. (We have also assumed that the prior probability on R is uniform over all sets of rankings, which is correct in the absence of any other rank information.)

The most likely values of the parameter functions α and β are now given by maximizing the marginal likelihood $P(G|\alpha,\beta) = \sum_R P(G,R|\alpha,\beta)$, or equivalently maximizing its logarithm, which is more convenient. The logarithm satisfies the Jensen inequality

$$\log \sum_R P(G,R|\alpha,\beta) \geq \sum_R q(R) \log \frac{P(G,R|\alpha,\beta)}{q(R)}, \tag{5.2}$$

for any set of probabilities $q(R)$ such that $\sum_R q(R) = 1$, with the equality being recovered when

$$q(R) = \frac{P(G,R|\alpha,\beta)}{\sum_R P(G,R|\alpha,\beta)}. \tag{5.3}$$

This implies that the maximization of the log-likelihood on the left side of (5.2) is equivalent to the double maximization of the right side, first with respect to $q(R)$, which makes the right side equal to the left, and then with respect to α and β, which gives us the answer we are looking for. It may appear that expressing the problem as a double maximization in this way, rather than as the original single one, makes it harder, but in fact that's not the case.

The right-hand side of (5.2) can be written as $\sum_R q(R) \log P(G,R|\alpha,\beta) - \sum_R q(R) \log q(R)$, but the second term does not depend on α or β, so as far as α and β are concerned we need consider only the first term, which is simply the average $\overline{\mathcal{L}}$ of the log-likelihood over the distribution $q(R)$:

$$\overline{\mathcal{L}} = \sum_R q(R) \log P(G,R|\alpha,\beta). \tag{5.4}$$

Making use of Eq. (5.1) and neglecting an unimportant overall constant, we then have

$$\overline{\mathcal{L}} = \sum_R q(R) \sum_{i\neq j} \Big[\tfrac{1}{2} S_{ij} \log \alpha(r_i - r_j) + T_{ij} \log \beta(r_i - r_j) - \tfrac{1}{2}\alpha(r_i - r_j) - \beta(r_i - r_j)\Big], \tag{5.5}$$

where we have used the fact that $\alpha(r)$ is a symmetric function.

This expression can be simplified further. The first term in the sum is

$$\tfrac{1}{2}\sum_R q(R)\sum_{i\neq j} S_{ij}\log\alpha(r_i-r_j) = \tfrac{1}{2}\sum_z\sum_{i\neq j} S_{ij}q(r_i-r_j=z)\log\alpha(z), \tag{5.6}$$

where $q(r_i - r_j = z)$ means the probability within the distribution $q(R)$ that $r_i - r_j = z$. We can define

$$a(z) = \frac{1}{n-|z|}\sum_{i\neq j} S_{ij}q(r_i-r_j=z), \tag{5.7}$$

which is the expected number of undirected edges in the observed network between pairs of nodes with rank difference z. It is the direct equivalent in the observed network of the quantity $\alpha(z)$, which is the expected number of edges in the model. The quantity $a(z)$, like $\alpha(z)$, is necessarily symmetric, $a(z) = a(-z)$, and hence (5.6) can be written as

$$\tfrac{1}{2}\sum_R q(R)\sum_{i\neq j} S_{ij}\log\alpha(r_i-r_j) = \sum_{z=1}^{n-1}(n-z)a(z)\log\alpha(z). \tag{5.8}$$

Similarly, we can define

$$b(z) = \frac{1}{n-|z|}\sum_{i\neq j} T_{ij}q(r_i-r_j=z) \tag{5.9}$$

and

$$\sum_R q(R)\sum_{i\neq j} T_{ij}\log\beta(r_i-r_j) = \sum_{z=1}^{n-1}(n-z)\big[b(z)\log\beta(z)+b(-z)\log\beta(-z)\big], \tag{5.10}$$

where $b(z)$ is the expected number of directed edges between a pair of nodes with rank difference z. Our final expression for $\overline{\mathcal{L}}$ is

$$\begin{aligned}\overline{\mathcal{L}} &= \sum_{z=1}^{n-1}(n-z)\bigl[a(z)\log\alpha(z) - \alpha(z) \\ &\qquad + b(z)\log\beta(z) - \beta(z) + b(-z)\log\beta(-z) - \beta(-z)\bigr]. \quad (5.11)\end{aligned}$$

Our approach involves maximizing this expression with respect to $\alpha(z)$ and $\beta(z)$ for given $a(z)$ and $b(z)$, which can be done using standard numerical methods. (Note that the expression separates into terms for the directed and undirected edges, so the two can be maximized independently.) The values of $a(z)$ and $b(z)$ in turn are calculated from Eqs. (5.3), (5.7), and (5.9), leading to an iterative method in which we first guess values for $\alpha(z)$ and $\beta(z)$, use them to calculate $q(R)$ and hence $a(z)$ and $b(z)$, then maximize $\overline{\mathcal{L}}$ to derive new values of α and β, and repeat to convergence. This is the classic expectation–maximization approach to model fitting.

To put this scheme into practice we need to specify a parameterization for the functions α and β, so that we can represent them on the computer. Since there are only a fixed number of values that the rank difference z can take in Eq. (5.11) [$2(n-1)$ of them, to be precise] we could in principle represent the functions completely by specifying their value separately for each z. This would, however, probably overfit the data because in reality the functions must be quite smooth—we do not expect small differences in rank to make a big difference to the probability of friendship. For a smooth function, a simple and natural parameterization is to use a Fourier series, which is the choice we make in this chapter. (We have experimented with other smooth parameterizations and find that our main conclusions are robust to the choice made.) The Fourier series on its own, however, turns out to be inadequate to completely represent the probability functions. As discussed in the following section, we find that a substantial fraction of friendships in the networks analyzed in this chapter run between individuals with closely similar rank, and we have found it necessary, in order to get a good fit to the model, to incorporate this observation into the parameterization by adding an additional term to both α and β consisting of a Gaussian peak of variable width, centered at the origin. With this addition we achieve robust fits to the model that are consistent across networks.

For the parameterization of the function α, which describes the probability of reciprocated friendship, we find good fits with only the central Gaussian peak plus a small uniform constant, which one can think of as the zeroth term in the Fourier expansion. For β, which represents the unreciprocated friendships, a more complicated form is needed—we use a five-term Fourier cosine series plus the central Gaussian peak. To some extent the choice of five terms is dictated by what's computationally feasible—our current numerical framework limits us to the five used here, but with improved computational resources it is possible that one could include more terms and achieve better fits. Nonetheless the fits achieved appear robust—as mentioned above, other parameterizations find similar functional forms.

Finally, we note that the sum in the denominator of Eq. (5.3) is too large to be tractable numerically. In the calculations presented here, therefore, we approximate it using a Markov chain Monte Carlo method—we generate complete rankings R in proportion to the probability $q(R)$ given by Eq. (5.3) and average over them to calculate $a(z)$ and $b(z)$.

5.3 Results

We have applied the method of the previous section to the analysis of data from the US National Longitudinal Study of Adolescent Health (the "AddHealth" study), a large-scale multi-year study of social conditions for school students and young adults in the United States. Using results from surveys conducted in 1994 and 1995, the study compiled friendship networks for over 90 000 students in schools covering US school grades 7 to 12 (ages 12 to 18 years). Schools were chosen to represent a broad range of socioeconomic conditions. High schools (grades 9 to 12) were paired with "feeder" middle schools (grades 7 and 8) so that networks spanning schools could be constructed.

To create the networks, each student was asked to select, from a list of students attending the same middle/high school combination, up to ten people with whom they were friends, with a maximum of five being male and five female. (Students were also asked to list their best friends first, but we ignore this in our analysis, treating all claimed friendships as equal.) From these selections, 84 friendship networks were constructed ranging in size from tens to thousands of students, one for each middle/high school pair, along with accompanying data on the participants, including school grade, sex, and ethnicity. Some of the networks divide into more than one strongly connected component, in which case we restrict our analysis to the largest component only. We perform the EM analysis of the previous section on each network separately, repeating the iterative procedure until the rankings no longer change.

Figure 5.1 shows results for a typical network. In panel (a), the histogram shows the measured value of the quantity $a(z)$, Eq. (5.7), the empirical probability of a reciprocated friendship (technically the expected number of undirected edges) between a vertex pair with rank difference z, with the horizontal axis rescaled to run from -1 to 1 (rather than $-n$ to n). As the figure shows the probability is significantly different from zero only for small values of z, with a strong peak centered on the origin. The solid curve shows the fit of this peak by the Gaussian function $\alpha(z)$, which appears good. The fit is similarly good for most networks. The form of $a(z)$ tells us that most reciprocated friendships fall between individuals of closely similar rank: there is a good chance that two people with roughly equal rank will both claim the other as a friend, but very little chance that two people with very different ranks will do so. This result seems at first surprising, implying as it does that people must be able to determine their own and others' rank with high accuracy in order to

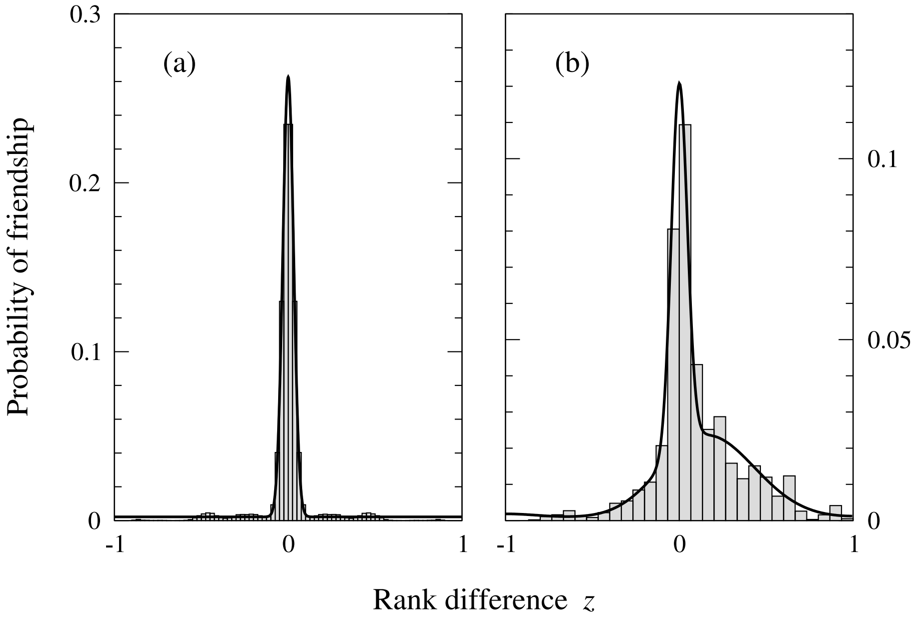


Figure 5.1: (a) Probability of reciprocated friendships as a function of rank difference (normalized to run from -1 to 1). The histogram shows empirical results for a single example network; the solid curve is the fitted function $\alpha(z)$. (b) The equivalent plot for unreciprocated friendships.

form friendships, but a number of previous studies have suggested that indeed this is true [7].

Panel (b) of Fig. 5.1 shows $b(z)$, Eq. (5.9), for the same network, which is the probability of a directed edge between nodes with rank difference z. Again there is a strong central peak to the distribution, of width similar to that for the undirected edges, indicating that many unreciprocated friendships are between individuals of closely similar rank. However, the distribution also has a substantial asymmetric tail for positive values of the rank difference, indicating that in a significant fraction of cases individuals claim friendship with those ranked higher than themselves, but that those claims are not reciprocated. The black curve in the panel shows the best fit to the function $\beta(z)$ in the maximum-likelihood calculation.

The general forms of these distributions are similar across networks from different schools. They also show interesting scaling behavior. The widths of the central peaks for both undirected and directed edges, when measured in terms of raw (unrescaled) rank difference are, to a good approximation, simply proportional to the average degree of a vertex in the network. Figure 5.2 shows these peaks for 78 of the 84 networks on two plots, for undirected edges (panel (a)) and directed edges (panel (b)), rescaled by average degree, and the approximately constant width is clear. (The six networks not shown are all small enough that the central peaks for the directed edges can be fit by the other parameters of the model and thus a direct comparison is not appropriate.) This result indicates that individuals have, roughly speaking, a fixed probability of being friends with others close to them in rank, regardless of the size of the community as a whole—as the average number of friends increases, individuals look proportionately further afield in terms of rank to find their friends, but are no more likely to be friends with any particular individual of nearby rank.

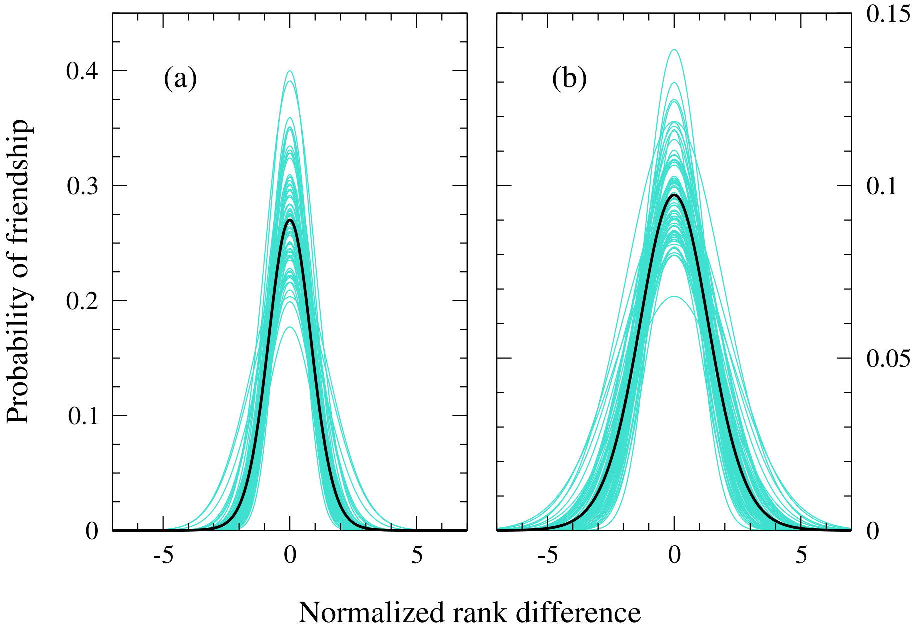


Figure 5.2: The fitted central peak of the friendship probability distributions for (a) reciprocated and (b) unreciprocated friendships. The horizontal axes are measured in units of absolute (unrescaled) rank difference divided by average network degree. Each blue curve is a network. The bold black curves represent the mean.

Outside of the central peak, i.e., for friendships between individuals with markedly different ranks, there are, to a good approximation, only unreciprocated friendships, and for these the shape of the probability distribution appears by contrast to be roughly constant when measured in terms of the rescaled rank of Fig. 5.1, which runs from -1 to 1. This probability, which is equal to the function $\beta(z)$ with the central Gaussian peak subtracted, is shown in Fig. 5.3 for the same 78 networks, rescaled vertically by the average probability of an edge to account for differing network sizes, and again the similarity of the functional form across networks is apparent, with low probability in the left half of the plot, indicating few claimed friendships with lower-ranked individuals, and higher probability on the right. The roughly constant shape suggests that, among the unreciprocated friendships, there is, for example, a roughly constant probability of the lowest-ranked student in the school claiming friendship with the highest-ranked, relative to other students, no matter how large the school may be.

The emerging picture of friendship patterns in these networks is one in which reciprocated friendships appear to fall almost entirely between individuals of closely similar rank. A significant fraction of the unreciprocated ones do the same, and moreover show similar scaling to their reciprocated counterparts, but the remainder seem to show a quite different behavior characterized by different scaling and by claims of friendship by lower-ranked individuals with substantially higher-ranked ones.

5.4 Analysis and Application

Taking the results of the previous section as a whole, we conjecture that the rankings discovered by the analysis correlate, at least approximately, with social status. If we assume that reciprocated friendships—almost all of which fall in the central peak—correspond to friendships in the conventional sense of mutual interaction, then a further conjecture, on the basis of similar statistics, is that the unreciprocated

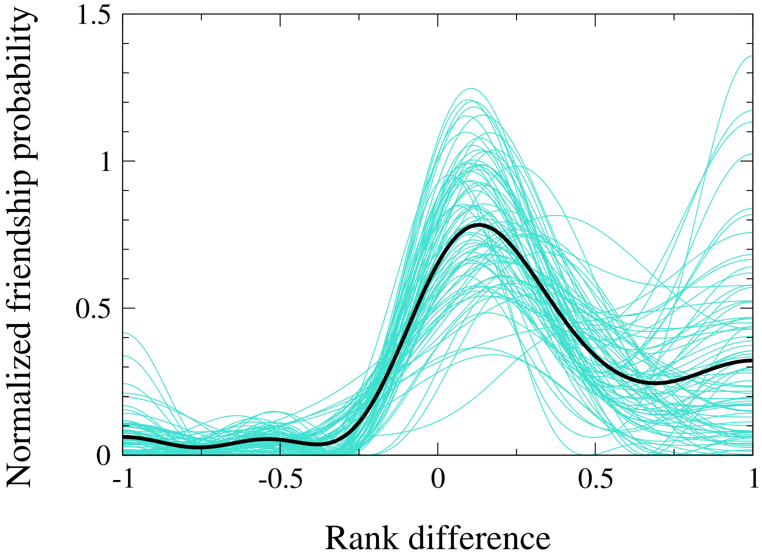


Figure 5.3: The fitted probability function for unreciprocated friendships, minus its central peak. The horizontal axis measures rank difference rescaled to run from −1 to 1. Each blue curve is a network. The bold black curve is the mean.

friendships in the central peak are also mutual but, for one reason or another, only one side of the relationship is represented in the data. One explanation why one side might be missing is that respondents in the surveys were limited to listing only five male and five female friends, and so might not have been able to list all of their friendships.

On the other hand, one might conjecture that the unreciprocated claims of friendship with higher-ranked individuals, those in the tail of the distribution in Fig. 5.1b, correspond to "aspirational" friendships, hopes of friendship with higher-ranked individuals that are, at present at least, not returned. Note also how the tail falls off with increasing rank difference: individuals are more likely to claim friendship with others of only modestly higher rank, not vastly higher.

One way to test these conjectures is to look for correlations between the rankings and other characteristics of individuals in the networks. For instance, it is generally thought that social status is positively correlated with the number of people who claim you as a friend [41, 65]. Figure 5.4a tests this by plotting average rank over all individuals in all networks (averaged in the posterior distribution of Eq. (5.1)) as a function of network in-degree (the number of others who claim an individual as a friend). As the figure shows, there is a strong positive slope to the curve, with the most popular individuals being nearly twice as highly ranked on average as the least popular. Figure 5.4b shows the corresponding plot for out-degree, the number of individuals one claims as a friend, and here the connection is weaker, as one might expect—claiming many others as friends does not automatically confer high status upon an individual—although the correlation is still statistically significant. Figure 5.4c shows rank as a function of total degree, in-degree plus out-degree, which could be taken as a measure of total social activity, and here again the correlation is strong. For all three panels the correlations are significant, with p-values less than 0.001.

In addition to the network structure itself, we have additional data about each of the participants, including their age (school grade), sex, and ethnicity. The distributions of rank for each sex and for individual ethnicities turn out to be close to uniform—a member of either sex or any ethnic group is, to a good approximation, equally likely to receive any rank from 1 to n, indicating that there is essentially no effect of sex or ethnicity on rank. (A Kolmogorov–Smirnov test does reveal deviations from uniformity in some cases, but the deviations are small, with KS statistics $D < 0.08$ in all instances.) Age, however, is a different story. Figure 5.5 shows the rescaled rank of individuals in each grade from 7 to 12, averaged over all individuals in all networks, and here there is a clear correlation. Average rank increases by more than a factor of two from the youngest students to the oldest (a one-way ANOVA gives $p < 0.001$). Since older students are generally acknowledged to have higher social status [32], this result lends support to the identification of rank with status. A further interesting wrinkle can be seen in the results for the 8th and 9th grades. Un-

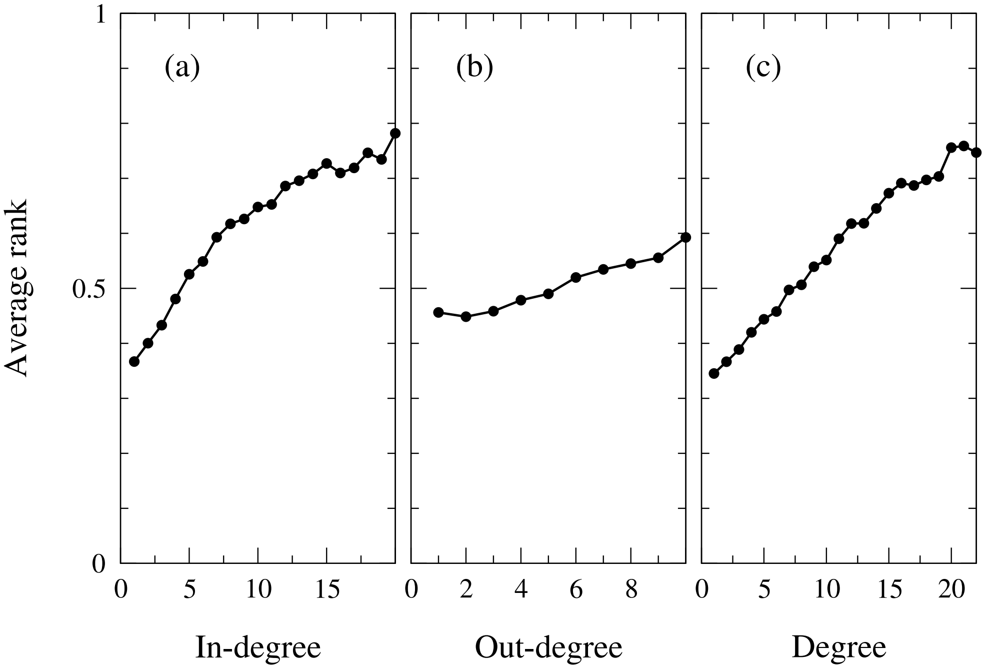


Figure 5.4: Plots of rescaled rank versus degree, averaged over all individuals in all networks for (a) in-degree, (b) out-degree, and (c) the sum of degrees. Measurement errors are comparable with or smaller than the sizes of the data points and are not shown.

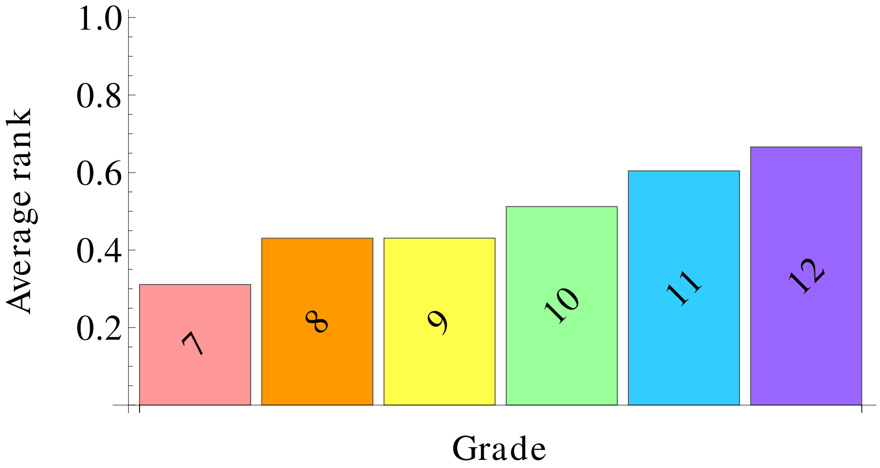


Figure 5.5: Rescaled rank as a function of school grade, averaged over all individuals in all schools.

like other pairs of consecutive grades, these two do not have a statistically significant difference in average rank (a t-test gives $p > 0.95$). This may reflect the fact that the 8th grade is the most senior grade in the feeder junior-high schools, before students move up to high school. When they are in the 8th grade, students are temporarily the oldest (and therefore highest status) students in school and hence may have a higher rank than would be expected were all students in a single school together.

Finally, in Fig. 5.6 we show an actual example of one of the networks, with nodes arranged vertically on a scale of inferred rank and colored according to grade. The increase of rank with grade is clearly visible, as is the fact that most undirected edges run between individuals of similar rank (and hence run horizontally in the figure).

5.5 Discussion

In this chapter, we have analyzed a large set of networks of friendships between students in American high and junior-high schools, focusing particularly on the distinction between friendships claimed by both participating individuals and friendships claimed by only one individual. We find that students can be ranked from low to high

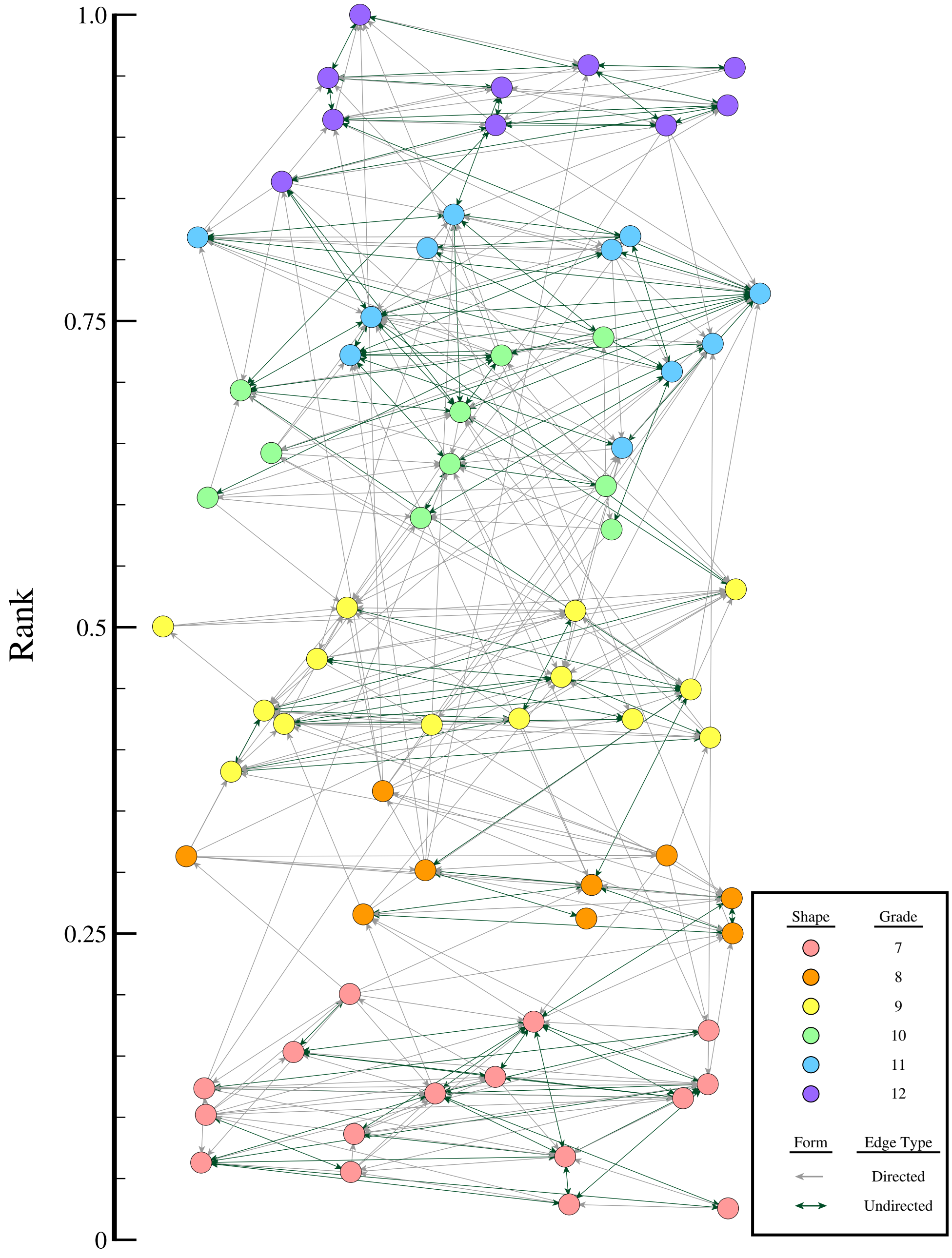


Figure 5.6: A sample network with (rescaled) rank on the vertical axis, vertices colored according to grade, and undirected edges colored differently from directed edges. Rank is calculated as an average within the Monte Carlo calculation (i.e., an average over the posterior distribution of the model), rather than merely the maximum-likelihood ranking. Note the clear correlation between rank and grade in the network.

such that most unreciprocated friendships consist of a lower-ranked individual claiming friendship with a higher-ranked one. We have developed a maximum-likelihood method for inferring such ranks from complete networks, taking both reciprocated and unreciprocated friendships into account, and we find that the rankings so derived correlate significantly with traditional measures of social status such as age and overall popularity, suggesting that the rankings may correspond to status. On the other hand, rankings seem to be essentially independent on average of other characteristics of the individuals involved such as sex or ethnicity.

There are a number of questions unanswered by our analysis. We have only limited data on the personal characteristics of participants. It would be interesting to test for correlation with other characteristics. Are rankings correlated, for instance, with academic achievement, number of siblings or birth order, number of Facebook friends, after-school activities, personality type, body mass index, wealth, or future career success? There is also the question of why a significant number of apparently close friendships are unreciprocated. One idea that has appeared in the literature is that some directed edges may correspond to new, temporary, or unstable friendships, which are either in the process of forming and will become reciprocated in the future, or will disappear over time [65, 134]. Evidence suggests that in practice about a half of the unreciprocated friendships do the former and a half the latter, and it is possible that the two behaviors correspond to the two classes of directed edges we identify in our analysis. A test of this hypothesis, however, would require longitudinal data—successive measurements of friendship patterns among the same group of individuals—data which at present we do not possess. Finally, there are potential applications of the statistical methods developed here to other directed networks in which direction might be correlated with ranking, such as networks of team or individual competition [114, 135] or dominance hierarchies in animal communities [36, 46].

CHAPTER VI

Conclusion

One of the techniques quickly achieving prominence in the study of networks is stochastic modeling. These network models are well-principled, based on probability and statistics rather than heuristics. This allows for a basic and detailed theoretical analysis that helps us to understand networks in general. Stochastic models are also flexible, allowing for a wide variety of structures. This dissertation has expanded our understanding of stochastic network models in several key areas, which also contributes to our general understanding of networks.

Chapter II introduces a stochastic blockmodel that focuses on undirected, unweighted overlapping community detection. This model does an excellent job finding communities in a network, competitive in accuracy with other modern algorithms. It also boasts a closed form expectation-maximization algorithm that allows for quick and efficient fitting. Several improvements are made to the algorithm, enabling it to run on even large modern networks in a reasonable amount of time. This model can also be used to find nonoverlapping communities in a principled fashion, as it relates to another stochastic blockmodel. With some simple tweaking of the resulting communities, it is possible to accurately detect nonoverlapping communities faster than previous methods for that particular model. In the future, it would be interesting to extend this model to allow for directed or weighted edges.

Chapter III extends the analysis of the BKN model presented in chapter II with a heuristic for determining the number of communities present in a network. The heuristic is based on likelihood ratio tests of the statistics literature. Applying the heuristic to networks works well at determining the number of communities, and this is demonstrated on both synthetic and real networks. In contrast, out-of-the-box methods fail in even simple tests. Additionally, the heuristic can do a decent job predicting the amount of overfitting when too many communities are being fit to a network generated according to the BKN model. Additional avenues for research include an explanation for the variance of the distributions, which the current heuristic does not account for. A stronger theoretical understanding of likelihood ratio tests for the model would also be ideal.

Chapter IV discusses a large multi-modal network taken from the Physical Review journals. Authors and papers are the two vertex types, and who authored which paper and which paper cited which other papers form the links in the network. When each paper was published is supplemental data that is used to study how scientific collaboration patterns have changed over time. It is discovered that scientists commonly cite themselves and their coauthors, and although there are concerns that this is becoming more prevalent recently to bolster citation indices, there does not appear to be any time dependence for the citation patterns in this dataset. Additionally, scientists cite themselves and their coauthors sooner than scientists they may be less familiar with. It is also common for researchers to return the favor of a citation to another researcher at a later time. Furthermore, transitivity is high among coauthors, so it is easy to see that there are strong social dynamics in with regard to publications. Surprisingly however, triadic closure only has a small (yet still significant) effect on the overall transitivity. Physical Review is the premier publication for Physics, but it would be nice to have additional journals to add to the dataset for a more thorough analysis. Another deficiency in the studies is taking subfields or even individual topics of Physics into account when comparing the numerical results to theoretical calculations. For example, two researchers studying networks are much more likely to collaborate or cite each other than they would with a high energy theorist.

Chapter V proposes a model for networks where the vertices have an inherent ranking. Vertices connect with other vertices in a semi-directed fashion based on the difference between their ranks and a connection function for directed and undirected edges. This model is applied to a set of high school friendship networks, and it appears that the ranks correspond with a measure of social status. Another observation is that students tend to reciprocate friendship with other students who have nearly the same social status, although many of the potential close friendships are not actually reciprocated. The remainder of the directed connections are reserved for students higher up the social ladder, and the result is a nearly acyclic network. These results hold across all networks in the dataset, which were taken from a wide variety of schools in the United States, showing that the dynamics observed are universal at least across this country. This model could also be applied to other types of networks with potentially different connection patterns, for example food webs or networks of competitive sports.

APPENDICES

APPENDIX A

Community detection and statistical text analysis

As mentioned in the main text, the generative model we use is the network equivalent of a model used in the text analysis technique called probabilistic latent semantic analysis (PLSA) [68, 69, 70], modified somewhat for the particular problem we are addressing. In this appendix, we describe PLSA and related methods and models and their relationship to the community detection problem.

A classic problem in text analysis, which is addressed by the PLSA method, is that of analyzing a "corpus" of text documents to find sets of words that all (or mostly) occur in the same documents. The assumption is that these sets of words correspond to topics or themes that can be used to group documents according to content. The PLSA approach regards documents as a so-called "bag of words," meaning one considers only the number of times each word occurs in a document and not the order in which words occur. (Also, one often considers only a subset of words of interest, rather than all words that appear in the corpus.)

Mathematically a corpus of D documents and W words of interest is represented by a matrix A having elements A_{wd} equal to the number of times word w appears in document d. To make the connection to networks, this matrix can be thought of as the incidence matrix of a weighted bipartite network having one set of vertices for the documents, one for the words, and edges connecting words to the documents in which they appear with weight equal to their frequency of occurrence.

In PLSA each word-document pair—an edge in the corresponding network picture—is associated with an unobserved variable u which denotes one of K topical groups. Each edge is assumed to be placed independently at random in the bipartite graph, with the probability that an edge falls between word w and document d being broken down in the form $\sum_u P(w|u)P(d|u)P(u)$, where $P(u)$ is the probability that the edge belongs to topic u, $P(w|u)$ is the probability that an edge with topic u connects to word w, and $P(d|u)$ is the probability that an edge with topic u connects to document d. Note that, given the topic, the document and word ends of each edge are placed independently. (Hofmann [68] calls this parameterization a "symmetric" one, meaning that the word and the document play equivalent roles mathematically, but in the networks jargon this would not be considered a symmetric formulation—the network is bipartite and the incidence matrix is not symmetric, nor even, in general, square.)

An alternative description of the model, which is useful for actually generating the incidence matrix and which corresponds with our formulation of the equivalent network problem, is that each matrix element A_{wd} takes a random value drawn independently from a Poisson distribution with mean $\sum_u P(w|u)P(d|u)\,\omega_u$. In the language of networks, each edge is placed with independent probability $\sum_u P(w|u)P(d|u)P(u)$, where $P(u) = \omega_u / \sum_{u'} \omega_{u'}$. In our work, where we focus on one-mode networks and a symmetric adjacency matrix instead of an incidence matrix, the parameter ω_u is redundant and we omit it.

PLSA involves using the edge probability above to calculate a likelihood for the entire word-document distribution, then maximizing with respect to the unknown probabilities $P(w|u)$, $P(d|u)$, and $P(u)$. The resulting probabilities give one a measure of how strongly each word or document is associated with a particular topic u, but since the topics are arbitrary, this is effectively the same as simply grouping the words and documents into "communities." Alternatively, one can use the probabilities to divide the edges of the bipartite graph among the topical groups, giving the text equivalent of the "link communities" that are the focus of our calculations.

A number of methods have been explored for maximizing the likelihood. Mathematically the one most closely related to our approach is the expectation-maximization (EM) algorithm of Hoffman [68, 69, 70], though the correspondence is not exact. Hofmann's work focuses solely on text processing—the connection to networks was not made until later—and because of its inherently asymmetric form the method cannot be translated directly for applications to standard one-mode networks. Instead we must reformulate the problem using a symmetric model, which leads to the approach described in chapter II. The symmetric formulation and the corresponding EM algorithm have not, to our knowledge, been used previously for community detection in networks, but several other related approaches have, including ones based on the techniques known as nonnegative matrix factorization (NMF) [44, 88] and latent Dirichlet allocation (LDA) [19, 57]. These formulations have similar goals to ours, but are typically asymmetric (and hence unsuitable for undirected networks) and use different algorithmic approaches for maximizing the likelihood. The NMF formulation is similar in style to an EM algorithm, using an iterative maximization scheme, but the specific iteration equations are different. Several papers have recently proposed using NMF to find overlapping communities [121, 141, 153], and in particular the work of Psorakis *et al.* [121] mentioned in the main text uses NMF with the PLSA model, although again in an asymmetric formulation, and not applied to link communities.

Recent work by Parkkinen *et al.* [115] and Gyenge *et al.* [64] does consider link communities, in an asymmetric formulation, but uses algorithmic approaches that are different again. For instance, Parkkinen *et al.* [115] use a model that attaches conjugate priors to the parameters and then samples the posterior distribution of link communities with a collapsed Gibbs sampler.

LDA [19, 57] offers an alternative but related approach that also attaches priors to the parameters, but in a specific way that relies on the asymmetric formulation of the model. In [66] and [154], LDA is adapted to networks by treating vertex-edge pairs as analogous to word-document pairs and then associating communities with the vertex-edge pairs. This is an interesting approach but differs substantially from the others discussed here, including our own, in which vertex-vertex pairs (i.e., edges) are the quantity analogous to word-document pairs.

Finally, in Appendix C we show that our model can be used to find nonoverlapping communities by viewing it as a relaxation of a nonoverlapping stochastic blockmodel. A corresponding relaxation has been noted previously for a version of NMF and was shown to be related to spectral clustering [42, 43].

APPENDIX B

Results for running time

As discussed in Section 2.4, a naive implementation of the EM equations gives an algorithm that is only moderately fast—not fast enough for very large networks. We described a more sophisticated implementation that prunes unneeded variables from the iteration and achieves significantly greater speed. In this appendix we give a comparison of the performance of the two versions of the algorithm on a set of test networks.

The results are summarized in Table B.1, which gives the CPU time in seconds taken to complete the overlapping community detection calculation on a standard desktop computer for each of the test networks. In these tests we use 100 random initializations of the variables and take as our final result the run that gives the highest value of the log-likelihood. For each network we give the results of three different calculations: (1) the calculation performed using the naive EM algorithm; (2) the calculation using the pruned algorithm with the threshold parameter δ set to zero, meaning the algorithm gives results identical to the naive algorithm except for numerical rounding, but runs faster; and (3) the calculation performed using the pruned algorithm with $\delta = 0.001$, which introduces an additional approximation that typically results in a slightly poorer final value of the log-likelihood, but gives a significant additional boost in speed.

The largest network studied, which is a network of links in the online community LiveJournal, is an exception to the pattern: for this network, which contains over 40 million edges, we performed runs with only ten random initializations each, using the pruned algorithm with $\delta = 0.001$ and with $\delta = 0$. Each randomly initialized run took about 50 minutes to complete for $\delta = 0.001$ and about 9 hours for $\delta = 0$.

While the algorithm described is fast by comparison with most other community detection methods, it is possible that its speed could be improved further (or that the quality of the results could be improved while keeping the speed the same). Two potential improvements are suggested by the text processing literature discussed in Appendix A. The first, from Hofmann [70], is to use the so-called tempered EM algorithm. The second, from Ding *et al.* [44], is to alternate between the EM algorithm and a nonnegative matrix factorization algorithm, exploiting the fact that both maximize the same objective function but in different ways.

Running conditions	Time (s)	Iterations	Log-likelihood
US air transportation, $n = 709$, $m = 3327$, $K = 3$			
naive, $\delta = 0$	15.71	55719	-8924.58
fast, $\delta = 0$	14.67	55719	-8924.58
fast, $\delta = 0.001$	2.17	26063	-9074.21
Network science collaborations [103], $n = 379$, $m = 914$, $K = 3$			
naive, $\delta = 0$	0.93	13165	-3564.74
fast, $\delta = 0$	0.82	13165	-3564.74
fast, $\delta = 0.001$	0.13	10747	-3577.85
Network science collaborations, $n = 379$, $m = 914$, $K = 10$			
naive, $\delta = 0$	3.19	18246	-2602.15
fast, $\delta = 0$	3.15	18246	-2602.15
fast, $\delta = 0.001$	0.49	12933	-2611.96
Network science collaborations, $n = 379$, $m = 914$, $K = 20$			
naive, $\delta = 0$	6.16	19821	-2046.95
fast, $\delta = 0$	6.09	19821	-2046.95
fast, $\delta = 0.001$	0.94	14010	-2094.85
Political blogs [1], $n = 1490$, $m = 16\,778$, $K = 2$			
naive, $\delta = 0$	11.42	13773	-48761.1
fast, $\delta = 0$	11.46	13773	-48761.1
fast, $\delta = 0.001$	4.14	13861	-48765.6
Physics collaborations [101], $n = 40\,421$, $m = 175\,693$, $K = 2$			
naive, $\delta = 0$	4339.57	424077	-1.367×10^6
fast, $\delta = 0$	2557.91	424077	-1.367×10^6
fast, $\delta = 0.001$	253.41	61665	-1.378×10^6
Amazon copurchasing [89], $n = 403\,394$, $m = 2\,443\,408$, $K = 2$			
naive, $\delta = 0$	170646.9	1222937	-2.521×10^7
fast, $\delta = 0$	105042.3	1222937	-2.521×10^7
fast, $\delta = 0.001$	11635.0	120612	-2.538×10^7
LiveJournal [8, 90], $n = 4\,847\,571$, $m = 42\,851\,237$, $K = 2$			
fast, $\delta = 0$	333230	278707	-4.611×10^8
fast, $\delta = 0.001$	33924	19257	-4.642×10^8

Table B.1: Example networks and running times for each of the three versions of the overlapping communities algorithm described in the text. The designations "fast" and "naive" refer to the algorithm with and without pruning respectively. "Iterations" refers to the total number of iterations for the entire run, not the average number for one random initialization. "Time" is similarly the total running time for all initializations. Directed networks were symmetrized for these tests. All networks were run with 100 random initializations, except for the LiveJournal network, which was run with 10 random initializations. Calculations were run on one core of a 4-core 3.2 GHz Intel Core i5 CPU with 4 GB memory under the Red Hat Enterprise Linux operating system. Running times do not include the additional cluster aggregation process described in Section 2.5.2, but in practice the extra time for this process is negligible.

APPENDIX C

Nonoverlapping communities

In Section 2.6 we described a procedure for extracting nonoverlapping community assignments from network data by first finding overlapping ones and then assigning each vertex to the community to which it belongs most strongly. This procedure was presented as a heuristic strategy for the nonoverlapping problem, but in this appendix we show that it can be derived in a principled manner as an approximation method for fitting the data to a degree-corrected stochastic blockmodel.

Methods have been proposed for discovering nonoverlapping communities in networks by fitting to the class of models known as stochastic blockmodels. As discussed in Ref. [83], it turns out to be crucial that the blockmodel used incorporate knowledge of the degree sequence of the network if it is to produce useful results, and this leads us to consider the so-called degree-corrected blockmodel, which can be formulated as follows. We consider a network of n vertices, with each vertex belonging to exactly one community. The community assignments are represented by an indicator variable s_{iu} which takes the value 1 if vertex i belongs to community u and zero otherwise. To generate the network, we place a Poisson distributed number of edges between each pair of vertices i, j, such that the expected value of the adjacency matrix element A_{ij} is $\theta_i \omega_{uv} \theta_j$ if vertex i belongs to group u and vertex j belongs to group v, where θ_i and

ω_{uv} are parameters of the model. To put this another way, the expected value of the adjacency matrix element is $\theta_i\bigl(\sum_{uv} s_{iu}\omega_{uv}s_{jv}\bigr)\theta_j$ for every vertex pair. The normalization of the parameters is arbitrary, since we can rescale all θ_i by the same constant if we simultaneously rescale all ω_{uv}. In our calculations we fix the normalization so that the θ_i sum to unity within each community: $\sum_i \theta_i s_{iu} = 1$ for all u.

Now one can fit this model to an observed network by writing the probability of generation of the network as a product of Poisson probabilities for each (multi-)edge, then maximizing with respect to the parameters θ_i and ω_{uv} and the community assignments s_{iu}. Unfortunately, while the maximization with respect to the continuous parameters θ_i and ω_{uv} is a simple matter of differentiation, the maximization with respect to the discrete variables s_{iu} is much harder. A common way around such problems is to "relax" the discrete variables, allowing them to take on continuous real values, so that the optimization can be performed by differentiation. In the present case, we allow the s_{iu} to take on arbitrary non-negative values, subject to the constraint that $\sum_u s_{iu} = 1$. In effect, s_{iu} now represents the fraction by which vertex i belongs to group u, with the constraint ensuring that the fractions add correctly to 1.

With this relaxation, we can now absorb the parameters θ_i into the s_{iu}, defining $\theta_{iu} = \theta_i s_{iu}$ with $\sum_i \theta_{iu} = 1$, and the mean number of edges between vertices i and j becomes $\sum_{uv} \theta_{iu}\omega_{uv}\theta_{jv}$. This is an extended form of the overlapping communities model studied in chapter II, generalized to include the extra $K \times K$ matrix ω_{uv}. In the language of link communities, this generalization gives us a model in which the two ends of an edge can belong to different communities. One can think of each end of the edge as being colored with its own color, instead of the whole edge taking only a single color. If ω_{uv} is constrained to be diagonal, then we recover the single-color version of the model again.

We can fit the general (nondiagonal) model to an observed network using an expectation-maximization algorithm, just as before. Defining a probability $q_{ij}(u, v)$

that an edge between i and j has colors u and v, the EM equations are now

$$q_{ij}(u,v) = \frac{\theta_{iu}\omega_{uv}\theta_{jv}}{\sum_{uv}\theta_{iu}\omega_{uv}\theta_{jv}}, \tag{C.1}$$

and

$$\theta_{iu} = \frac{\sum_{jv} A_{ij} q_{ij}(u,v)}{\sum_{ijv} A_{ij} q_{ij}(u,v)}, \qquad \omega_{uv} = \sum_{ij} A_{ij} q_{ij}(u,v). \tag{C.2}$$

By iterating these equations we can find a solution for the parameters θ_{iu}. But $\theta_{iu} = \theta_i s_{iu}$ and, summing both sides over u, we get $\sum_u \theta_{iu} = \theta_i$, since $\sum_u s_{iu} = 1$. Hence

$$s_{iu} = \frac{\theta_{iu}}{\theta_i} = \frac{\theta_{iu}}{\sum_u \theta_{iu}}. \tag{C.3}$$

Thus we can calculate the values of s_{iu} and once we have these we can then reverse the relaxation of the model by rounding the values to zero or one, which is equivalent to assigning each vertex i to the community u for which s_{iu} is largest, or equivalently the community for which θ_{iu} is largest.

Thus the final algorithm for dividing the network is simply to iterate the EM equations to convergence and then assign each vertex to the community for which θ_{iu} is largest. In the language of Section 2.6, this is equivalent to looking for the largest value of k_{iu}/κ_u, and hence this algorithm is the same as the algorithm that we described in that section, except that the model is generalized to include the matrix ω_{uv}, where in our original calculations this matrix was absent, which is equivalent to assuming it to be diagonal. In our experiments, however, we have found that even when we allow ω_{uv} to be nondiagonal, the algorithm usually chooses a diagonal value anyway, which implies that the output of our original algorithm and the generalized algorithm should be the same. (We note that in practice the diagonal version of the algorithm runs faster, while both are substantially faster than the vertex moving heuristic proposed for the stochastic blockmodel in Ref. [83].)

Diagonal values are expected for networks with traditional community structure, where connections are more dense within communities than between them. It is entirely possible, however, that there could be networks with interesting nondiagonal group structure that could be detected using the more general model. The model including the matrix ω_{uv} can in principle find disassortative community structure—structure in which connections are less common within communities than between them—as well as the better studied assortative structure. For example, it can detect bipartite structure in networks, whereas the unadjusted model cannot.

APPENDIX D

Attainability of the maximal solution

In this appendix, we'll prove that the global maximum of a network must be attainable for the overlapping community detection blockmodel of the text. Strictly speaking, for our formulation to be a properly defined likelihood function, A_{ij} must be an integer, but we can easily define an extension function

$$f(A'|\mathbf{\Theta};K) = \sum_{ij} A'_{ij} \log\bigl(\textstyle\sum_u \theta_{iu}\theta_{ju}\bigr) - \sum_{iju} \theta_{iu}\theta_{ju}, \tag{D.1}$$

where A'_{ij} is allowed to be any nonnegative number. Set $A'_{ij} = A_{ij} + \epsilon$ where ϵ is some small perturbation. Trivially, $f(A|\mathbf{\Theta};K) = \log P(A|\mathbf{\Theta};K)$.

Now we make two choices. First, choose ϵ small enough that as we deform it down to 0, the position of the maximum of the function doesn't change discontinuously. This is a valid choice since the function is smooth with respect to all the parameters. Next, notice that $\sum_u \theta_{iu}\theta_{ju} > 0$ for each pair of i, j for every valid parameter set – in particular the set that absolutely maximizes f – since $A'_{ij} \geq \epsilon > 0$. In fact, this is the only constraint on the $\mathbf{\Theta}$, although it should also be noted that there is a rotational symmetry to the system: given a rotation $\mathbf{R}$, $f(A'|\mathbf{\Theta};K) = f(A'|\mathbf{\Theta'};K)$ where $\mathbf{\Theta'}$ is defined by setting $\theta'_{\mathbf{i}} = \mathbf{R}\theta_{\mathbf{i}}$. Due to this rotational symmetry, we can always choose the maximum of f where each vector $\theta_{\mathbf{i}}$ lies in the first sectant (i.e. $\theta_{iz} \geq 0$ for all i and z).

Using the two facts from the previous paragraph together, limit ϵ down to 0. The first sectant is a closed set, so the limit solution $\theta_{\mathbf{i}}$ must also be contained in the first sectant. This parameter set maximizes $f(A|\mathbf{\Theta}; K)$, which we chose specifically to be $\log P(A|\mathbf{\Theta}; K)$. Thus the absolute maximum of $\log P(A|\mathbf{\Theta}; K)$ is always attainable.

The importance of this argument to the application of Wilks' Theorem to the overlapping community detection model is rather subtle. If the maximum weren't always attainable, then we would be completely sunk and would have no hope of solving for the change in log-likelihood with the general approach of likelihood ratio tests. However, since we know the best attainable solution has to be a valid global maximum too, the worst that can happen is that some parameters will degenerate and not count towards the distribution.

APPENDIX E

Data processing

As mentioned in the main text, we performed some pre-processing on the raw Physical Review data to disambiguate author names and remove extreme outliers. This appendix describes the steps taken.

The data were supplied in two blocks: (1) a list of papers with associated information, such as authors, author affiliation, journal, and year of publication; (2) a list of citations, using unique paper identifiers that correspond to entries in the first block. There are, however, no unique identifiers for authors that are consistent between papers, making unambiguous author identification difficult. Not all authors use the same form for their name on every publication, and there are many examples of distinct researchers with the same name. Before using the data set, therefore, we made an effort to associate names of authors with unique people. As in previous work on author disambiguation, our process starts by assuming every name on every paper to represent a different individual [122], then computes a number of measures of author similarity and assumes authors who are sufficiently similar by these measures to be the same person. After completing this disambiguation process we checked a subset of the results by hand to estimate error rates for the process and found that it performs well. Details are as follows.

Our approach relies not only on the author names themselves to establish similarity, but also on collaboration patterns and institutional affiliation, since authors with similar names who have many of the same collaborators or who are at the same institution are more likely to be the same person. Affiliation information, however, like the author names themselves, tends to be ambiguous and inconsistent, so our first step is to combine affiliations that are deemed similar enough. We measure similarity using a variant of edit distance applied to the affiliation text strings, implemented using the Python difflib library.

Once the affiliations are processed in this way, we process the author names as follows:

1. We combine all authors with identical names who share an institutional affiliation. It appears to be uncommon for two physicists at the same university to publish under identical names, so this seems to be a safe step.

2. We find author pairs with similar but not identical names. Our criterion for similarity at this stage is that authors should have identical last names and compatible first/middle names (i.e., identical if fully written out, or compatible initials where initials are used). Also authors should not have published together on the same paper (which rules out, for example, family members with similar names who publish together). For all pairs with similar names we then calculate a further similarity measure based on how many affiliations they share, how many coauthors they share, whether their full names are identical, and whether they have published in the same journal. Authors with a high enough similarity are combined, most similar pairs first.

We have tested the accuracy of this process by drawing two lists at random from its output, the first containing 79 instances in which authors with similar names have been combined into a single author, and the second containing 111 instances in which they have not. We then performed, by hand, a blind search—without knowing the choice the algorithm has made—for publicly available on-line information about the names in question, to determine whether they do indeed represent the same or distinct researchers. We find the false positive rate to be 3% (i.e., 3% of pairs are incorrectly judged to be the same person when in reality they are distinct) and the false negative rate to be 12%.

We also tested the effect on our results of the disambiguation process by calculating a number of the statistics reported in this paper both for the disambiguated data and for the raw data set before disambiguation, in which we naively assume that every unique author string represents a unique author and every pair of authors with the same string are the same person. We found substantial differences between the two in some of the most basic statistics, such as total number of distinct authors: the number was 328 938 in the raw data set, but fell to 235 533 after disambiguation. On the other hand some other statistics changed very little, indicating that these are not particularly sensitive to details of author identification. For example, the clustering coefficient changes from 0.222 in the raw data set to 0.212 in the disambiguated data set.

In addition to author disambiguation we cull the data according to a few simple rules. There are a number of papers in the data set that have no authors listed, primarily editorials and other logistical articles without scientific content. These we remove entirely. As mentioned in the text, we also identify all papers with fifty or more coauthors, and many of our calculations are performed in two versions, with and without these papers. The choice of fifty authors as the cutoff point was made by inspection of the distribution of author numbers shown in Fig. E.1. As the figure shows, the number of papers with a specific number of coauthors appears, roughly speaking, to follow a power law (in agreement with some previous studies [100], but not others [76]), but there is a marked deviation from the power-law form for the high-

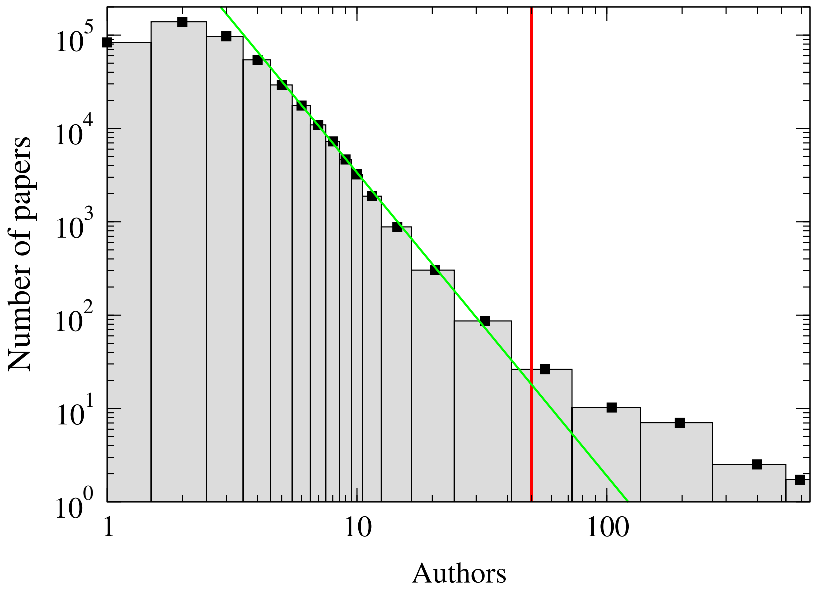


Figure E.1: Histogram of the number of papers with a given number of authors. The vertical line falls at fifty authors and corresponds roughly to the point at which the distribution deviates from the power-law form indicated by the fit. The data for ten authors and more have been binned logarithmically to minimize statistical fluctuations.

est numbers of coauthors, above about fifty, indicating potentially different statistical laws in this regime, and possibly different underlying collaborative processes.

We also removed from the data a small number of citations. In a few cases a paper is listed as citing itself, which we assume to be an error. In a number of other cases papers cite others that were published at a later time, which violates causality. These too are assumed to be erroneous and are removed. Finally, the data indicate that some papers cited the same other paper several times within the one bibliography; such multiple citations we count as a single citation.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Lada A. Adamic and Natalie Glance. The political blogosphere and the 2004 US election. pages 36–43, 2005.

[2] Yong-Yeol Ahn, James P. Bagrow, and Sune Lehmann. Link communities reveal multi-scale complexity in networks. *Nature*, 466:761–764, 2010.

[3] Edoardo M. Airoldi, David S. Choi, and Patrick J. Wolfe. Confidence sets for network structure. *Statistical Analysis and Data Mining*, 4(5):461–469, 2011.

[4] Edoardo M. Airoldi, David M. Blei Stephen E. Fienberg, and Eric P. Xing. Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.*, 9:1980–2014, 2008.

[5] H. Akaike. A new look at the statistical identification model. *IEEE Trans. Auto. Control*, 19:716–723, 1974.

[6] Iqbal Ali, Wade D. Cook, and Moshe Kress. On the minimum violations ranking of a tournament. *Management Science*, 32:660–672, 1986.

[7] Cameron Anderson, Sanjay Srivastava Jennifer S. Beer, Sandra E. Spataro, and Jennifer A. Chatman. Knowing your place: Self-perceptions of status in face-to-face groups. *Journal of Personality and Social Psychology*, 91:1094–1110, 2006.

[8] Lars Backstrom, Dan Huttenlocher, Jon Kleinberg, and Xiangyang Lan. Group formation in large social networks: Membership, growth, and evolution. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 44–54, 2006.

[9] James P Bagrow. Evaluating local community methods in networks. *J. Stat. Mech.*, 2008.

[10] Brian Ball, Brian Karrer, and M.E.J. Newman. Efficient and principled method for detecting communities in networks. *Phys. Rev. E*, 84, 2011.

[11] Brian Ball and M.E.J. Newman. Friendship networks and social status. *Network Science*, 1:16–30, 2013.

[12] David L. Banks and Kathleen M. Carley. Models for network evolution. *Journal of Mathematical Sociology*, 21:173–196, 1996.

[13] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[14] Marc Barthélemy. Spatial networks. *Physics Reports*, 499:1–101, 2011.

[15] Christoph Bartneck and Servaas Kokkelmans. Detecting h-index manipulation through self-citation analysis. *Scientometrics*, 87(1):85–98, 2011.

[16] Edward A. Bender and E. Rodney Canfield. The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory A*, 24:296–307, 1978.

[17] Peter J. Bickel and Aiyou Chen. A nonparametric view of network models and newman–girvan and other modularities. *Proc. Natl. Acad. Sci. USA*, 106(50):21068–21073, 2009.

[18] Peter J. Bickel and Purnamrita Sarkar. Hypothesis testing for automated community detection in networks. Preprint arxiv:1311.2694, 2013.

[19] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.

[20] Mindaugas Bloznelis and Valentas Kurauskas. Clustering function: a measure of social influence. Technical report.

[21] Ulrik Brandes, Daniel Delling, Marco Gaertler, Robert Görke, Martin Hoefer, Zoran Nikoloski, and Dorothea Wagner. On modularity clustering. *IEEE Transactions on Knowledge and Data Engineering*, 20:172–188, 2008.

[22] Andrei Broder, Ravi Kumar andFarzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, and Janet Wiener. Graph structure in the web. *Computer Networks*, 33:309–320, 2000.

[23] Hal Burch and Bill Cheswick. Internet mapping project. `http://www.cheswick.com/ches/map/index.html`.

[24] Rodrigo de Castro and Jerrold W. Grossman. Famous trails to Paul Erdős. *The Mathematical Intelligencer*, 21:51–53, 1999.

[25] Hanfeng Chen and Jiahua Chen. The likelihood ratio test for homogeneity in finite mixture models. *Canadian Journal of Statistics*, 29(2):201–215, 2001.

[26] Hanfeng Chen, Jiahua Chen, and John D. Kalbfleisch. A modified likelihood ratio test for homogeneity in finite mixture models. 63(1):19–29, 2001.

[27] P. Chen and S. Redner. Community structure of the physical review citation network. *Journal of Informetrics*, 4:278–290, 2010.

[28] Fan Chung and Linyuan Lu. The average distances in random graphs with given expected degrees. *Proceedings of the National Academy of Science*, 99(25), 2002.

[29] Fan R. K. Chung. *Spectral Graph Theory. Number 92 in CBMS Regional Conference Series in Mathematics.* American Mathematical Society, Providence, RI, 1997.

[30] Aaron Clauset, Cosma Rohilla Shalizi, and M. E. J. Newman. Power-law distributions in empirical data. *SIAM Rev.*, 51:661–703, 2009.

[31] J. E. Cohen. Ecologists' co-operative web bank, version 1.0: machine-readable data base of food webs. 1989.

[32] James S. Coleman. *The Adolescent Society: The Social Life of the Teenager and its Impact on Education.* Greenwood Press, 1961.

[33] Leon Danon, Jordi Duch, Albert Diaz-Guilera, and Alex Arenas. Comparing community structure identification. *J. Stat. Mech.*, 2005.

[34] Jörn Davidsen, Holger Ebel, and Stefan Bornholdt. Emergence of a small world from local interactions: Modeling acquaintance networks. *Phys. Rev. Lett.*, 88, 2002.

[35] James A. Davis and Samuel Leinhardt. The structure of positive interpersonal relations in small groups. *Sociological Theories in Progress*, 2:218–251, 1972.

[36] Han de Vries. Finding a dominance order most consistent with a linear hierarchy: a new procedure and review. *Animal Behaviour*, 55:827–843, 1998.

[37] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Phys. Rev. E*, 84, 2011.

[38] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Inference and phase transitions in the detection of modules in sparse networks. *Phys. Rev. Lett.*, 107, 2011.

[39] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B*, 39(1):1–38, 1977.

[40] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[41] Jan Kornelis Dijkstra, Antonius H. N. Cillessen, Siegwart Lindenberg, and René Veenstra. Basking in reflected glory and its limits: Why adolescents hang out with popular peers. *Journal of Research on Adolescents*, 20(4):942–958, 2010.

[42] Chris Ding, Xiaofeng He, and Horst D. Simon. On the equivalence of nonnegative matrix factorization and spectral clustering. pages 606–610, 2005.

[43] Chris Ding, Tao Li, and Michael I. Jordan. Nonnegative matrix factorization for combinatorial optimization: Spectral clustering, graph matching, and clique finding. pages 183–192, 2008.

[44] Chris Ding, Tao Li, and Wei Peng. On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing. *Comput. Stat. Data Anal.*, 52:3913–3927, 2008.

[45] Patrick Doreian, Vladimir Batagelj, and Anuska Ferligoj. Symmetric-acyclic decompositions of networks. *Journal of Classification*, 17:3–28, 2000.

[46] Carlos Drews. The concept and definition of dominance in animal behaviour. *Behaviour*, 125:283–313, 1993.

[47] P. Erdős and A. Rényi. On random graphs. *Publicationes Mathematicae*, 6:290–297, 1959.

[48] P. Erdős and A. Rényi. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, 5:17–61, 1960.

[49] P. Erdős and A. Rényi. On the strength of connectedness of a random graph. *Acta Mathematica Academiae Scientiarum Hungarica*, 12:261–267, 1961.

[50] T. S. Evans and R. Lambiotte. Line graphs, link partitions, and overlapping communities. *Phys. Rev. E*, 80, 2009.

[51] Jianqing Fan and Heng Peng. Nonconcave penalized likelihood with a diverging number of parameters. *Annals of Statistics*, 32(3), 2004.

[52] M. Fiedler. Algebraic connectivity of graphs. *Czech. Math. J.*, 23:298–305, 1973.

[53] L. R. Ford, Jr. and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956.

[54] Santo Fortunato. Community detection in graphs. *Phys. Rep.*, (486):75–174, 2010.

[55] Santo Fortunato and Marc Barthélemy. Resolution limit in community detection. *Proc. Natl. Acad. Sci. USA*, 104(1):36–41, 2007.

[56] M. T. Gastner and M. E. J. Newman. The spatial structure of networks. *Eur. Phys. J. B*, 49:247–252, 2006.

[57] Mark Girolami and Ata Kabán. On an equivalence between plsi and lda. pages 433–434, 2003.

[58] Michelle Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proc. Nat. Acad. Sci. USA*, 99:7821–7826, 2002.

[59] Benjamin H. Good, Yves-Alexandre de Montjoye, and Aaron Clauset. Performance of modularity maximization in practical contexts. *Phys. Rev. E*, 81, 2010.

[60] Jerrold W. Grossman. The evolution of the mathematical research collaboration graph. *Congressus Numerantium*, 158:201–212, 2002.

[61] Jerrold W. Grossman and Patrick D. F. Ion. On a portion of the well-known collaboration graph. *Congressus Numerantium*, 108:129–131, 1995.

[62] S. Gualdi, M. Medo, and Y.-C. Zhang. Influence, originality and similarity in directed acyclic graphs. *Europhysics Letters*, 96(1), 2011.

[63] Roger Guimerà, Marta Sales-Pardo, , and Luís A. Nunes Amaral. Modularity from fluctuations in random graphs and complex networks. *Phys. Rev. E*, 70, 2004.

[64] Ádám Gyenge, Janne Sinkkonen, and András A. Benczúr. An efficient block model for clustering sparse graphs. pages 62–69, 2009.

[65] Maureen T. Hallinan and Warren N. Kubitschek. The effect of individual and structural characteristics on intransitivity in social networks. *Social Psychology Quarterly*, 51(2):81–92, 1988.

[66] Keith Henderson and Tina Eliassi-Rad. Applying latent dirichlet allocation to group discovery in large graphs. pages 1456–1461, 2009.

[67] J. E. Hirsch. An index to quantify an individuals scientific research output. *Proceedings of the National Academy of Science*, 102(46):16569–16572, 2005.

[68] Thomas Hofmann. Probabilistic latent semantic indexing. *Proceedings of the 22nd Annual International ACM Conference on Research and Development in Information Retrieval*, pages 50–57, 1999.

[69] Thomas Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Mach. Learn.*, 42:177–196, 2001.

[70] Thomas Hofmann. Latent semantic models for collaborative filtering. *Mach. Learn.*, 22:89–115, 2004.

[71] Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109 – 137, 1983.

[72] Paul W. Holland and Samuel Leinhardt. A method for detecting structure in sociometric data. *American Journal of Sociology*, 76:492–513, 1970.

[73] Paul W. Holland and Samuel Leinhardt. An exponential family of probability distributions for directed graphs. *Journal of the American Statistical Association*, 76(373):62 – 65, 1981.

[74] Petter Holme and Beom Jun Kim. Growing scale-free networks with tunable clustering. *Phys. Rev. E*, 65, 2002.

[75] George Caspar Homans. *The Human Group*. Harcourt, Brace, 1950.

[76] Jiann-Wien Hsu and Ding-Wei Huang. Distribution for the number of coauthors. *Phys. Rev. E*, 80, 2009.

[77] J. Huang, Z. Zhuang, J. Li, and C. L. Giles. Collaboration over time: characterizing and modeling network evolution. *Proceedings of the International Conference on Web Search and Web Data Mining*, pages 107–116, 2008.

[78] Kerson Huang. *Statistical Mechanics*. Wiley, Hoboken, NJ, 1987.

[79] Peter J. Huber. Robust regression: asymptotics, conjectures and monte carlo. *Annals of Statistics*, 1(5):799–821, 1973.

[80] Emily M. Jin, Michelle Girvan, and M. E. J. Newman. Structure of growing social networks. *Phys. Rev. E*, 64, 2001.

[81] Richard M. Karp. Reducibility among combinatorial problems. pages 85–103, 1972.

[82] Brian Karrer and M. E. J. Newman. Random graph models for directed acyclic networks. *Phys. Rev. E*, 80(4), 2009.

[83] Brian Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. *Phys. Rev. E*, 83(1), 2011.

[84] B. W. Kernighan and S. Lin. An efficient heuristic procedure for partitioning graphs. *Bell System Technical Journal*, 49:291–307, 1970.

[85] D. E. Knuth. *The Stanford GraphBase: A Platform for Combinatorial Computing*. Addison–Wesley, Reading, MA, 1993.

[86] Andrea Lancichinetti and Santo Fortunato. Community detection algorithms: A comparative analysis. *Phys. Rev. E*, 80, 2009.

[87] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. *Phys. Rev. E*, 78, 2008.

[88] Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[89] Jure Leskovec, Lada A. Adamic, and Bernardo A. Huberman. The dynamics of viral marketing. *ACM Trans. Web*, 1(1), 2007.

[90] Jure Leskovec, Kevin J. Lang, Anirban Dasgupta, and Michael W. Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):1–123, 2009.

[91] Alfred J. Lotka. The frequency distribution of scientific productivity. *Journal of the Washington Academy of Sciences*, 16:317–324, 1926.

[92] Travis Martin, Brian Ball, Brian Karrer, and M.E.J. Newman. Coauthorship and citation patterns in the physical review. *Phys. Rev. E*, 88, 2013.

[93] K. Menger. Zur allgemeinen kurventheorie. *Fundamenta Mathematicae*, 10:96–115, 1927.

[94] Stanley Milgram. The small world problem. *Psychology Today*, 2:60–67, 1967.

[95] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298:824–827, 2002.

[96] Michael Molloy and Bruce Reed. A critical point for random graphs with a given degree sequence. *Random Structures and Algorithms*, 6:161–179, 1995.

[97] James Moody. Race, school integration, and friendship segregation in america. *American Journal of Sociology*, 107(3), 2001.

[98] J. L. Moreno. *Shall Survive?: Foundations of Sociometry, Group Psychotherapy, and Sociodrama.* Beacon House, Beacon, NY, 1934.

[99] M. E. J. Newman. Clustering and preferential attachment in growing networks. *Phys. Rev. E*, 64, 2001.

[100] M. E. J. Newman. Scientific collaboration networks: I. network construction and fundamental results. *Phys. Rev. E*, 64, 2001.

[101] M. E. J. Newman. The structure of scientific collaboration networks. *Proc. Nat. Acad. Sci. USA*, 98(2):404–409, 2001.

[102] M. E. J. Newman. The structure and function of complex networks. *SIAM Rev.*, 45:167–256, 2003.

[103] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E*, 74, 2006.

[104] M. E. J. Newman. The first-mover advantage in scientific publication. *Europhys. Lett.*, 86(6), 2009.

[105] M. E. J. Newman. Random graphs with clustering. *Phys. Rev. Lett.*, 103, 2009.

[106] M. E. J. Newman. *Networks: An Introduction.* Oxford University Press, Inc., 2010.

[107] M. E. J. Newman, Albert-László Barabási, and Duncan J. Watts. *The Structure and Dynamics of Networks.* Princeton University Press, Princeton, NJ.

[108] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69, 2004.

[109] M. E. J. Newman and E. A. Leicht. Mixture models and exploratory analysis in networks. *Proc. Nat. Acad. Sci. USA*, 104, 2007.

[110] M. E. J. Newman and Juyong Park. Why social networks are different from other types of networks. *Phys. Rev. E*, 68, 2003.

[111] M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E*, 64, 2001.

[112] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435:814–818, 2005.

[113] Raj Kumar Pan, Kimmo Kaski, and Santo Fortunato. World citation and collaboration networks: uncovering the role of geography in science. *Scientific Reports*, 2(902), 2012.

[114] Juyong Park and M. E. J. Newman. A network-based ranking system for American college football. *J. Stat. Mech.*, 2005.

[115] Juuso Parkkinen, Janne Sinkkonen, Adam Gyenge, and Samuel Kaski. A block model suitable for sparse graphs. 2009.

[116] R. K. Pathria. *Statistical Mechanics.* Academic Press, Waltham, MA, 1972.

[117] Derek de Solla Price. A general theory of bibliometric and other cumulative advantage processes. *Journal of the American Society for Information Science*, 27:292–306, 1976.

[118] Derek J. de Solla Price. *Science since Babylon.* Yale University Press, New Haven, 1961.

[119] Derek J. de Solla Price. *Little Science, Big Science.* Columbia University Press, New York, 1963.

[120] Derek J. de Solla Price. Networks of scientific papers. *Science*, 149(3683):510–515, 1965.

[121] Ioannis Psorakis, Stephen Roberts, and Mark Ebden. Overlapping community detection using bayesian non-negative matrix factorization. *Phys. Rev. E*, 83, 2011.

[122] F. Radicchi, S. Fortunato, B. Markines, and A. Vespignani. Diffusion of scientific credits and the ranking of scientists. *Phys. Rev. E*, 80, 2009.

[123] Anatol Rapoport. Cycle distributions in random nets. *The bulletin of mathematical biophysics*, 10:145–157, 1948.

[124] Anatol Rapoport and William J. Horvath. A study of a large sociogram. *Behavioral Science*, 6:279–291, 1961.

[125] Sidney Redner. Citation statistics from 110 years of physical review. *Physics Today*, 58:49–54, 2005.

[126] Gerhard Reinelt. *The Linear Ordering Problem: Algorithms and Applications.* Heldermann, Berlin, 1985.

[127] Christian P. Robert and George Casella. *Monte Carlo Statistical Methods.* Springer, Berlin, 1999.

[128] Soma Sanyal. Effect of citation patterns on network structure. 2007.

[129] Gideon Schwarz. Estimating the dimension of a model. 6(2):461–464, 1978.

[130] William Shockley. On the statistics of individual variations of productivity in research laboratories. *Proceedings of the IRE*, 45:279–290, 1957.

[131] Tom A. B. Snijders and Krzysztof Nowicki. Estimation and prediction for stochastic blockmodels for graphs with latent block structure. *Journal of Classification*, 14(1):75–100, 1997.

[132] Tom A.B. Snijders. Statistical models for social networks. *Annual Review of Sociology*, 37:131 – 153, 2011.

[133] Ray Solomonoff and Anatol Rapoport. Connectivity of random nets. *Bulletin of Mathematical Biophysics*, 13:107–117, 1951.

[134] Aage B. Sørensen and Maureen T. Hallinan. A stochastic model for change in group structure. *Social Science Research*, 5(1):43–61, 1976.

[135] Raymond T. Stefani. Survey of the major world sports rating systems. *Journal of Applied Statistics*, 24:635–646, 1997.

[136] Frank Strauss and Michael Ikeda. Pseudolikelihood estimation for social networks. *Journal of the American Statistical Association*, 85(409):204 – 212, 1990.

[137] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58:267–288, 1996.

[138] R. Wagner-Döbler. Continuity and discontinuity of collaboration behaviour since 1800 — from a bibliometric point of view. *Scientometrics*, 52:503–517, 2001.

[139] Matthew L. Wallace, Vincent Larivière, and Yves Gingras. Modeling a century of citation distributions. *Journal of Informetrics*, 3:296–303, 2009.

[140] Matthew L. Wallace, Vincent Larivière, and Yves Gingras. A small world of citations? the influence of collaboration networks on citation practices. *PLoS ONE*, 7, 2012.

[141] Fei Wang, Tao Li, Xin Wang, Shenghuo Zhu, and Chris Ding. Community discovery using nonnegative matrix factorization. *Data Mining and Knowledge Discovery*, 22:493–521, 2011.

[142] Yuchung J. Wang and George Y. Wong. Stochastic blockmodels for directed graphs. *Journal of the American Statistical Association*, 82(397):8–19, 1987.

[143] Stanley Wasserman and Katherine Faust. *Social Network Analysis*. Cambridge University Press, Cambridge, 1994.

[144] Stanley Wasserman and Philippa Pattison. Logit models and logistic regressions for social networks: I. an introduction to markov graphs and p*. *Psychometrika*, 60(3):401–425, 1996.

[145] Duncan J. Watts. *Six Degrees: The Science of a Connected Age*. Norton, New York, NY, 2003.

[146] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.

[147] J. G. White, E. Southgate, J. N. Thomson, and S. Brenner. The structure of the nervous system of the nematode caenorhabditis elegans. *Philosophical Transactions of the Royal Society of London*, 314(1165):1–340, 1986.

[148] Allen W. Wilhite and Eric A. Fong. Coercive citation in academic publishing. *Science*, 335(6068):542–543, 2012.

[149] S. S. Wilks. The large-sample distribution of the likelihood ratio for testing composite hypotheses. *Ann. Math. Statist.*, 9(1):60–62, 1938.

[150] S. Wuchty, B. Jones, and B. Uzzi. The increasing dominance of teams in production of knowledge. *Science*, 316(5827):1036–1039, 2007.

[151] Xiaoran Yan, Jacob E. Jensen, Florent Krzakala, Cristopher Moore, Cosma Rohilla Shalizi, Lenka Zdeborova, Pan Zhang, and Yaojia Zhu. Model selection for degree-corrected block models. (arxiv:1207.3994), 2012.

[152] W. W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452–473, 1977.

[153] Mina Zarei, Dena Izadi, and Keivan Aghababaei Samani. Detecting overlapping community structure of networks based on vertex–vertex correlations. *J. Stat. Mech.*, 2009.

[154] Haizheng Zhang, Baojun Qiut, C. Lee Giles, Henry C. Foley, and John Yen. An lda-based community structure discovery approach for large-scale social networks. pages 200–207, 2007.

[155] X. S. Zhang, R. S. Wang, Y. Wang, J. Wang, Y. Qiu, L. Wang, and L. Chen. Modularity optimization in community detection of complex networks. *Europhys. Lett.*, 87, 2009.

[156] Han Zhu, Xinran Wang, and Jian-Yang Zhu. Effect of aging on network structure. *Phys. Rev. E*, 68, 2003.