Review of Statistical Network Analysis: Models, Algorithms and Software

M. Salter-Townshend[*], A. White, I. Gollini and T.B. Murphy[†‡]

22 March 2012

Abstract

The analysis of network data is an area that is rapidly growing, both within and outside of the discipline of statistics.

This review provides a concise summary of methods and models used in the statistical analysis of network data, including the Erdős-Renyi model, the exponential family class of network models and recently developed latent variable models. Many of the methods and models are illustrated by application to the well-known Zachary karate dataset. Software routines available for implementing methods are emphasised throughout.

The aim of this paper is to provide a review with enough detail about many common classes of network model to whet the appetite and to point the way to further reading.

[*]Corresponding Author

[†]Clique Strategic Research Cluster, School of Mathematical Sciences & Complex and Adaptive Systems Laboratory, University College Dublin, Dublin 4, Ireland.

[‡]This material is based upon work supported by the Science Foundation Ireland under Grant No. 08/SRC/I1407: Clique: Graph & Network Analysis Cluster

1 Introduction

Network modelling has long been of interest to statisticians but in recent years it has begun to rapidly evolve, attracting growing interest both within statistics and more widely. A network is any dataset that is composed of actors or nodes upon which we have relational data. These relationships may be directed or undirected; they may be or discrete or continuous and are often continuous but measured as discrete.

In recent times, analyses of network data have created headlines in both the peer-reviewed academic literature and in mainstream media. [33], for example, concluded that "people were most likely to become obese when a friend became obese" in an analysis based on a large thirty two year social network study; this analysis received widespread publicity in the mainstream media including the New York Times. Subsequent studies by the same researchers have also claimed that network effects are important in smoking [32], happiness [44] and alcohol consumption [95], although these findings are not without controversy [80]. Furthermore, the onset of Web 2.0 social networking sites such as Facebook and Linkedin have allowed the comparatively easy collection of huge contemporary social network datasets [e.g. 108]. Network analysis has also become a popular method of investigation in the biological sciences in terms of protein-protein, gene-gene and gene-protein interaction networks. Indeed "analysis of protein-protein interaction (PPI) networks has become a major thrust in systems biology research" [120]. Other applications of networks analysis include terrorist networks [71], sexually transmitted infections [37], financial fraud [99] and co-citation networks [78], to name but a few.

In this paper we present a succinct review up to and including the current state of the art in statistical network modelling. Section 2 provides an introduction to the terminology and concepts from graph theory which are commonly used in network analysis. Network visualisation methods are discussed in Section 3. Some of the classical probabilistic models are outlined in Section 4 and more complex latent variable models are introduced in Section 5. Goodness-of-fit and model validation methods are briefly discussed in Section 6. In the concluding Section 7 we outline some of the challenges that lie ahead. In an accompanying supplement we provide results of all models applied to a larger Facebook network along with the code used to create the figures and results presented, along with some not included in this paper due to space constraints.

For simplicity of exposition, we confine our study to models for static binary network data; thus the relationship between nodes is that either a link (edge) is present or it is not, with relationships only considered at one occasion (static). While more complex network data exists, we note that the network models developed for such data are (thus far) extensions of the models discussed in this paper. We cite

the literature in cases where such developments exist.

Other reviews and collections that give excellent overviews of social network analysis and statistical modelling of network data include [116], [26], [118], [3] and [50]. While these offer summaries of statistical network modelling that are possibly more comprehensive than our own, we offer this document as an alternative and more concise paper with a focus on brevity and on the relationships between various statistical network models. This paper is intended to serve as a contemporary text to provide introductory level material, with appropriate references. We note that statistics is only one of many fields in which network analysis is a topic of interest. A wide range of non-probabilistic models are available in the computer science, machine learning and other literature; this review concentrates on the statistical modelling of network data rather than on deterministic approaches.

1.1 Zachary's Karate Club

We demonstrate ideas and examples throughout the paper using the well known Zachary's Karate club data [122]. This dataset was chosen for familiarity and for compatibility with a wide range of the methods reviewed in this paper. While the network is discussed in more detail in subsequent sections, here we provide some context. The dataset consists of the friendship network of 34 members of a university based karate club. It was collected following observations over a three year period in which a factional division caused the members of the club to formally separate into two organisations. While friendships within the network arose organically, a financial dispute regarding the pay of part time instructor Mr. Hi tested these ties, with two political factions developing. Key to the dispute were two members of the network, the aforementioned Mr. Hi and club president John A. When the dispute eventually led to the dismissal of Mr. Hi by John A., his supporters resigned from the karate club and established a new organisation, headed by Mr. Hi. The dataset exhibits many of the phenomenon observed in sociological studies, namely the development of communities and emergence of prominent figures. It is thus ideal for the demonstration of statistical models which seek to identify such behaviours in a quantifiable manner.

2 Graph Theory & Descriptive Statistics

In this section we provide a brief review of the common terminology and notation used in the literature, as well as a description of empirical network statistics.

2.1 Basic Definitions and Measures

A network is typically described using the language of graph theory. [1] A graph $\mathcal{G}(\mathcal{N}, \mathcal{L})$ consists of a set of N nodes (or vertices) $\mathcal{N} = \{n_1, n_2, \ldots, n_N\}$ and a set of L edges (or connections) $\mathcal{L} = \{l_1, l_2, \ldots, l_L\}$, which denotes the links between nodes. An adjacency or socio-matrix Y, of dimension $N \times N$ can also be used to represent $\mathcal{G}$, with

$$y_{ij} = \begin{cases} 1 & \text{edge exists from node } n_i \text{ to node } n_j \\ 0 & \text{otherwise} \end{cases}$$

Note that this applies only for a binary network. Weighted (or valued) networks are described using non-negative integer values for the entries in Y. Binary networks are also expressible using an *edge list* E. This is a two column matrix with edge l given by the l^{th} row of E and where e_{l1} is the node index of the source of edge l and e_{l2} is the destination node of the edge.

Reflexivity, Symmetry These terms refer to behaviour common within many observed real world networks. A node n_i is *reflexive* if it is adjacent to itself, that is if $y_{ii} = 1$. This may or may not be possible in a network, depending on the data in question. In friendship data, for example, an actor's relationship with themselves is typically ignored or treated as undefined, whereas in protein-protein interaction data, proteins are often capable of self-interaction. Ties within a network may be *symmetric* or *reciprocal*. For example, in many instances we expect ties of friendship to be returned, or mathematically, that between two nodes n_i and n_j, we have $y_{ij} = y_{ji}$. The graph representing an *asymmetric* network is said to be directed, in which case it may be referred to as a digraph. A symmetric network is also referred to as undirected.

Transitivity A third commonly observed phenomenon is that of *transitivity*. This may be loosely interpreted as "the friend of my friend is my friend." More formally, a graph displays transitivity if nodes n_i and n_j being connected and nodes n_j and n_k being connected implies that node n_i is likely to be connected to node n_k. That is, if $y_{ij} = 1$ and $y_{jk} = 1$ then this implies that $P(y_{ik} = 1)$ is greater than if y_{ij} and y_{jk} are zero. There is no rigid definition of when a graph is transitive, rather there are continuous measurements of transitivity such as the ratio of number of triangles to connected triples in the graph.

[1] We do not discuss in detail the relationship between estimating graphical models and network analysis, apart from noting that much of the terminology and statistics overlap. For example, estimation of a sparse Gaussian graphical model by regularization of the inferred precision (inverse covariance) matrix using lasso type methods is the subject of some recent research (see for example [46] and [83]). These methods seek to establish which variables in a model are dependent and the resulting graph resembles a network. However, this topic is not to be confused with network analysis where the links between nodes are the data.

Geodesic Distance, Connectedness and Diameter Graph theory also provides tools to measure the connectivity structure of a network. The *geodesic distance* $d(i,j) = \min_k y_{ij}^{[k]} > 0$ denotes the degree of separation, or length of the shortest path, between nodes n_i and n_j, where $y_{ij}^{[k]} = 1$ if there is a path of length k between nodes i and j. If the geodesic distance is finite for all nodes in the network, the graph is said to be *connected*. Otherwise, the graph is *unconnected*.

When the graph is unconnected, it can be decomposed into components, or maximally connected sub-graphs. The *diameter* of a graph is defined to be the largest geodesic distance (i.e. the length of the longest shortest path) between any two nodes in the network.

Using this terminology, we can describe the Karate Club dataset in more detail. The data is an undirected graph, consisting of 34 irreflexive nodes and 78 undirected edges. The diameter of the Zachary Karate Club is 5 and a path of this length is highlighted in Figure 2(a). While our dataset is connected, this is in fact because it is the largest component of a larger dataset, with many members disengaged from the political in-fighting within the group, and excluded for this reason [122].

2.2 Network Summary Statistics

While the set of edges $\mathcal{L}$ measures the interaction between all actors in the network $\mathcal{N}$, we may be interested in how actors interact at a more local level. This behaviour within the graph can be summarised using various statistics; these are particularly relevant to the models discussed in Sections 4.2 and 4.4.

The simplest way to quantify a node's connectivity is to consider the number of nodes with which it is incident. The *degree* of a node n_i in an undirected graph is the number of edges incident with the node, so that $d(n_i) = y_{i+} = \sum_j y_{ij} = y_{+i} = \sum_{j=1} y_{ji}$. For a directed graph, the *indegree* is given by $d_i(n_i) = y_{+i}$ and *outdegree* by $d_o(n_i) = y_{i+}$.

Next, we consider a network at the dyadic level. This is where we only consider the interaction between two actors in the network. A directed graph consisting of two nodes can take one of three states: *mutual* when $y_{ij} = y_{ji} = 1$, *asymmetric* when $y_{ij} = 1$ and $y_{ji} = 0$, or vice-versa, or *null* when $y_{ij} = y_{ji} = 0$. Counting how often these states occur across the entire network is referred to as the *dyad census*.

In order to study higher degree network effects, particular types of sub-graph are of interest, such as k-cliques (a sub-graph of k nodes where all nodes are connected to each other), or k-stars (a sub-graph of $k+1$ nodes in which k of the nodes are connected through a single node) — of particular interest are 3-cliques, or triangles, which illustrate transitivity at its most local level. Examples of these graph structures are shown in Figure 1.

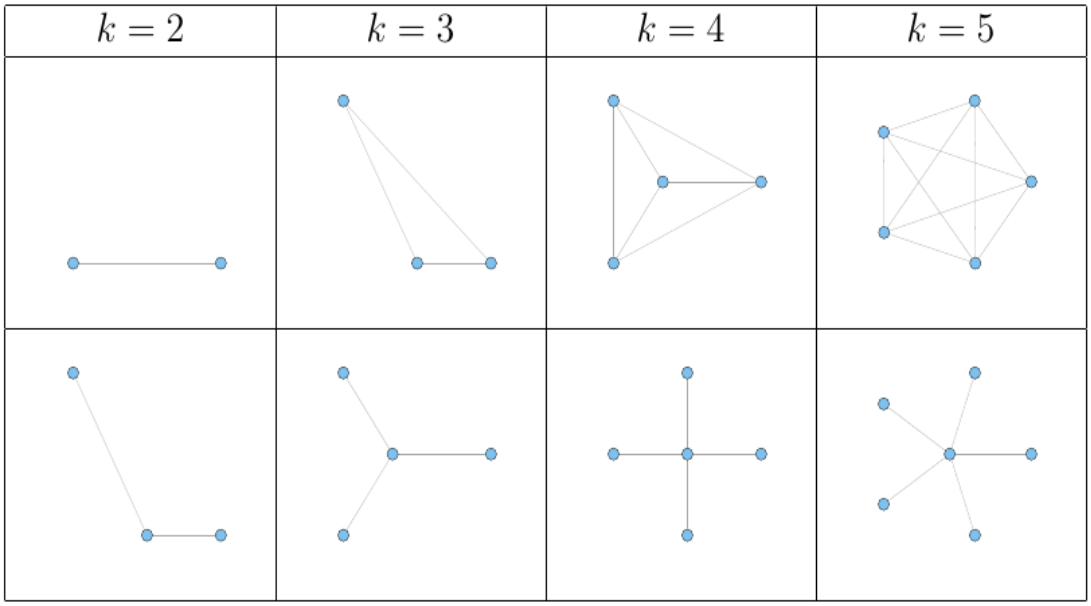


Figure 1: Examples of k-cliques (top row) and k-stars (bottom row). Note that typically the k in k-cliques and in k-stars refer to the number of nodes and edges involved respectively. That is, a k-star involves $k + 1$ nodes.

The total number of triangles in a graph, $T(Y)$, can be found directly by calculating $T(Y) = \sum_{i \leq j \leq k} y_{ij} y_{jk} y_{kl}$. Obtaining other summary statistics is less straightforward. [65] define statistics $D_0(Y), \ldots, D_{N-1}(Y)$ and $P_0(Y), \ldots, P_{N-2}(Y)$ as the *degree distribution* and *partner distribution* of Y respectively. $D_i(Y)$ is defined to be the number of nodes in the network Y with degree equal to i. $P_i(Y)$ is defined to be the number of dyads which are incident with each other and share exactly i neighbours in common. Several statistics of interest may then be directly computed using D_i and P_i. For example, let $S_k(Y)$ and $T(Y)$ denote the number of k-stars and triangles in the graph Y respectively. Then

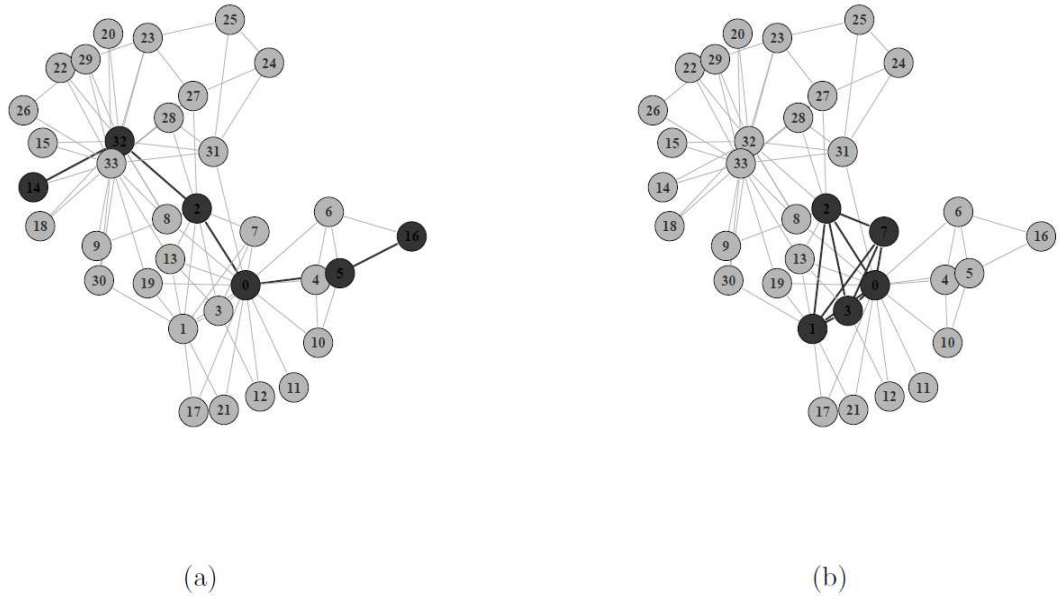
$$S_k(Y) = \sum_{i=1}^{N-1} \binom{i}{k} D_i(Y) \text{ for } k \geq 2$$

and

$$T(Y) = \frac{1}{3} \sum_{i=1}^{N-2} i P_i(Y).$$

Applied to the karate dataset, we note that Mr. Hi was a member of both 5-cliques in the network. There are also 43 triangles, in comparison with 77 edges, indicating the high level of transitivity within the group.

For several more examples of summary statistics, see [93] and [116].



(a) (b)

Figure 2: Examples of some of the definitions used in graph theory. (a) shows the diameter of the Zachary Karate Club network. (b) shows the clique involving nodes $0, 1, 2, 3$ and 7, one of the two largest cliques in the Zachary network. See section 3 for more details on network visualisation.

2.3 Centrality and Prestige

Highlighting actors of importance to the network is a common task of statistical network analysis. *Centrality measures* are ways of representing this importance in a quantifiable way. A node (or actor)'s importance is considered to be the extent of their involvement in a network, and this can be measured in multiple ways. Centrality measures are usually applied to undirected networks, with indices for directed graphs termed *prestige measures*, although these methods are less developed [116]. Common centrality measures include:

Degree Centrality The simplest way to quantify a node's importance is to consider the number of nodes it is incident with, with high numbers interpreted to be of higher importance. Therefore the degree of a node provides local information of its importance. Since the maximum nodal degree is $N-1$, the standardised centrality measure is:

$$C_{D_o}(n_i) = \frac{y_{+i}}{(N-1)}.$$

Closeness Centrality Nodes can also be indexed by considering their geodesic distance to each other. The closeness centrality of node i is given by:

$$C_C(n_i) = \frac{N-1}{\sum_{j=1}^{N} d(n_i, n_j)}.$$

Note that the geodesic distance between two nodes in distinct components is undefined (or defined as infinite), and as such the closeness centrality measure is only meaningful for a connected network.

Betweeness Centrality Another way to gauge a node's influence is to consider its role in linking other nodes together in the network. Denoting σ_{jk} as the number of geodesics, or shortest paths, between nodes n_j and n_k, and let $\sigma_{jk}(n_i)$ be the number of geodesics containing node n_i. The betweeness centrality of node n_i is defined as

$$C_B(n_i) = \sum_{j<k} \frac{\sigma_{jk}(n_i)}{\sigma_{jk}} \frac{1}{\binom{N}{2}}.$$

A criticism of this measure is that it assumes that two nodes will be linked over only the shortest possible path, and that each geodesic is equally likely to be taken. Other, more complicated measures such as *information centrality* [106] attempt to reconcile this by considering the information contained in all paths containing a specific node.

Eigenvector Centrality Eigenvector centrality is another measure of centrality. The eigenvector centrality of each node can be found by computing the leading eigenvector of the adjacency matrix Y. This measure of centrality takes all connections in a network into account rather than just looking at shortest paths between nodes. The eigenvector centrality can be seen as an alternative to degree, where the connections are weighted according to node centralities [16]. A number of properties and interpretations are given in [17]. Eigenvector centrality is at the core of the PageRank [20] and the HITS [70] algorithms for ranking networks that we introduce in Section 2.4.2.

As an illustration, the application of these measures to the Zachary Karate Club network is shown in Figure 3. The five actors with highest centrality scores with respect to each measure are shown more darkly. Note that Mr. Hi and John A, nodes 0 and 33 respectively, are selected by each measure, underlining their importance within the network.

2.4 Community Finding, Clustering and Ranking

2.4.1 Community Finding and Clustering

Community finding and clustering methods are among the most common tasks of network analysis and are usually concerned with partitioning networks into highly connected sub-graphs. This is also referred to as community finding or clustering. The focus of this paper is on statistical and probabilistic methods for network analysis, and so we provide only a cursory summary of the deterministic methods for clustering.

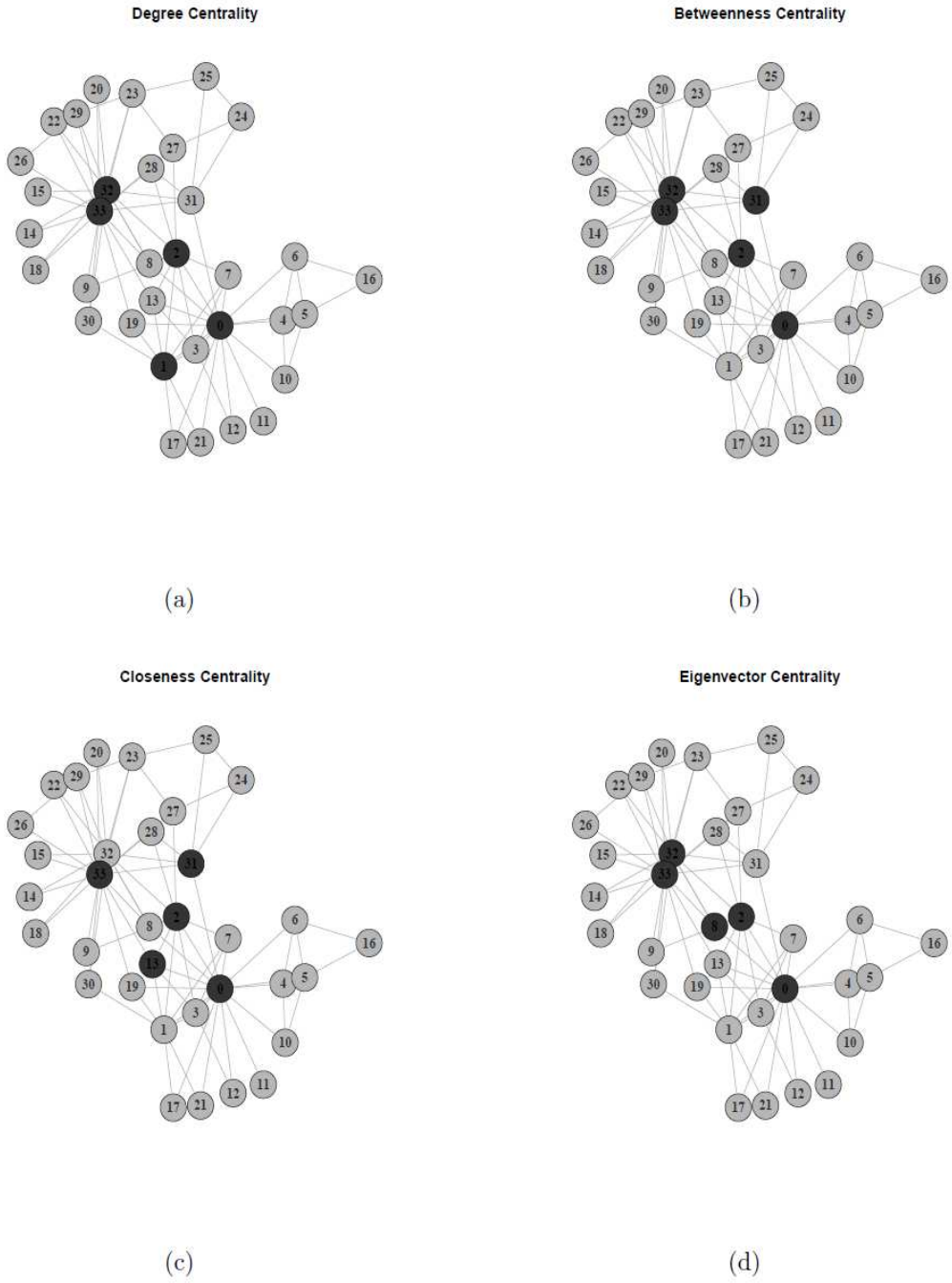


Figure 3: (a) Degree, (b) betweeness, (c) closeness and (d) eigenvector centrality measures applied to the Zachary Karate Club network, where the actors with the five highest centrality scores are shown more darkly. Note that nodes 0 and 33 are selected by each measure.

One of the first methods to make use of centrality measures for community detection was the Girvan-Newman clustering method [49]. The steps of the algorithm are simply:

1. Calculate betweenness for all edges in the network.

2. Remove the edge with highest betweenness.
3. Recalculate betweenness for all edges affected by the removal.
4. Repeat from step 2 until no edges remain.

By removing links with the highest betweenness centrality scores, and considering nodes who remain linked as being closer to one another than those then separated, a hierarchical clustering of the network becomes possible.

Other deterministic community finding methods have been proposed, such as spectral clustering [101, 85] which is based on an eigenvectors decomposition of the graph Laplacian. The Laplacian matrix is given by

$$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{Y}\mathbf{D}^{1/2}$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{D}$ is the diagonal matrix with the degree of the nodes on the diagonal and $\mathbf{Y}$ is the adjacency matrix. As per [85], in order to cluster into K components, the K largest eigenvectors are stacked as the columns in an $N \times K$ matrix. This matrix is then row-normalised to have unit length. Treating the rows as points in $\mathbb{R}^K$, a clustering routine such as k-means is used to cluster the nodes. See [113] for a tutorial on spectral clustering that compares and evaluates the more popular methods.

[119] note that such methods are biased towards clusters of equal size. [84] mitigated this problem when they introduced modularity based methods, which are amongst the most popular. Suppose that nodes in the network are partitioned into clusters, where c_i records the cluster membership of node n_i. The modularity [84] of the partitioning is defined as

$$Q = \frac{1}{2M} \sum_{\{(i,j):c_i=c_j\}} \left[y_{ij} - \frac{y_{i+}y_{j+}}{2M} \right],$$

where M is the number of edges in the graph. Many community (or cluster) finding algorithms aim to find the partition of the nodes into clusters that maximise the modularity [36], but it is a computationally challenging problem [18, 19]. Note that the use of modularity for community finding is not without criticism, in particular as they seek to divide the network into non-overlapping clusters [e.g. 43].

More recently, methods for finding overlapping communities (or clusters) have been developed [e.g. 87] including the CFinder algorithm [1] that uses k-cliques to construct overlapping communities.

2.4.2 Network Ranking

Network ranking problem is another interesting task in analyzing network data. One of the most used ranking methods is the PageRank algorithm presented by [20] which is at the basis of the Google search engine. In this context, the web pages

and the hyperlinks are respectively the nodes and the edges of the network. The PageRank (PR) of the node n_i is defined as

$$PR(n_i) = \frac{1-d}{N} + d\sum_{j=1}^{N} \frac{y_{ji}}{y_{j+}} PR(n_j)$$

where the parameter d, called the *damping factor*, takes values between 0 and 1. [20] suggest setting d equal to 0.85. The rank of a node then depends on the number of nodes from which they are refereed, weighted by their ranks. Many extensions to this method have been developed, and we refer to the survey papers [10, 98] for further details.

Another ranking method for networks is the Hypertext Induced Topics Search (HITS) algorithm [70]. This alghorithm assigns ranking values to nodes which identify both good link targets (*authority score*) and good link sources (*hub score*). The *authority score* for node n_i is defined as

$$a_i = \sum_{j=1}^{N} y_{ji} h_j$$

and the *hub score*

$$h_i = \sum_{j=1}^{N} y_{ij} a_j.$$

2.4.3 Semi-supervised Learning

Semi-supervised learning methods are useful tools for classification, ranking and link prediction problems. They use the information given by some labeled nodes in the network to estimate the unknown labels of the other nodes. Two main approaches for semi-supervised learning in graph are presented in [123, 124] and [25]. [123, 124] propose an algorithm which makes use of a kernel matrix based on dyadic links. This method has a good predictive power, but it is not able to handle very large networks due to high computational complexity. For that reason [25] proposed a new method called discriminative random walks ($\mathcal{D}$-walks) which allows to classify the unlabelled nodes according to a class betweenness measure that depends on the passage times during random walks performed in the input graph. Further details on semi-supervised learning can be found in [125] and [126]. New extensions to these methods for sparse large network have been recently proposed in [81].

2.5 Software

`igraph` [34] is a free open source software package that allows the efficient manipulation of directed and undirected graphs, potentially consisting of millions of nodes and edges. It is available to download in several formats, including as a package in

R [89]. `igraph` can perform various operations from graph theory, including calculating (at nodal level) the degree, closeness, betweenness and eigenvector centrality, and PageRank of a graph, decomposing a graph into its connected components, generating sub-graphs from a given graph, or generating random graphs using simple criteria. It also provides functions to calculate graph modularity and perform community finding algorithms, and can visualise graphs (see Section 3), both in two and three dimensions.

The `statnet` [53] suite of R packages, including `sna`, `ergm` and `network` [21, 54, 22], provide tools to obtain network summary statistics and both node and graph-level indices for a variety of centrality measures.

`spa` [35] is an R package to fit semi-parametric models for semi-supervised learning.

3 Visualisation

Visualisation plays a central role in both the exploratory and reporting stages of statistical analysis of networks and may even form the basis of a network model. Visualisation is of particular importance for exploration of small networks or networks with low rank structure. When dealing with larger problems, visualisation of the entire network usually leads to the "hairball" problem; the structure is too complex to project onto only two or even three dimensions and the edges overlap heavily. In this section we give a brief overview of some visualisation methods for network data.

3.1 Adjacency Matrix Visualisations

Direct visualisation of the adjacency matrix may be the simplest way to visualise a network. The rows and columns of the adjacency matrix are reordered such that the nodes are grouped into highly connected clusters; this is a form of seriation [5] of the adjacency matrix of the network. A heat-map plot of the matrix then demonstrates the degree to which the nodes are clustered. For example a network displaying no clustering will appear as a random matrix of dots whereas a highly clustered network with three clusters will appear as a matrix with three distinct and highly connected blocks about the diagonal and sparse connections between these blocks.

Alternatively, structural equivalence may be highlighted by ordering the nodes such that nodes grouped together, in the same block, relate to other nodes in a similar way. For an example of adjacency matrix visualisation, see Figure 4(a), and for further details on the theory on the blocking and structural equivalence, see Section 4.3.

3.2 Layout Algorithms

Many network visualisation methods consist of laying out the nodes on the plot and adding the links as line segments (or arrows) connecting the nodes . A layout algorithm is often chosen to minimise some criterion which tries to quantify the simplicity of the plot (e.g. the length of the links and the amount of crossover of the links in the plot).

Visualisation of large graphs is especially prone to "hairball" plots. That is, there are so many overlapping links that the nodes (and potentially interesting structure within the network) are completely obscured. However, even very small graphs can be subject to this issue if the layout algorithm is poorly chosen (see Figure 5(a), for example). Planar graphs are graphs that may be depicted without crossing links, but these are rare for anything but very small and sparse networks. Link routing layout algorithms seek to minimise the frequency of the link crossings.

Visualisation is used to find and to highlight structural properties in the graph. This may include clustering of nodes, etc. The selection of a layout algorithm appropriate to the individual network depends on which features of the dataset the user wishes to explore; for example colouring nodes by a particular attribute and using a layout algorithm that places connected nodes closer together than unconnected node pairs may highlight clustering-by-attribute. This section summarises some of the common algorithms.

There are broadly speaking five main methods of node layout for plotting networks (besides a random layout):

1. **Minimum cut:** the nodes in the network are laid out to minimise the number of edges that cross each other. Results may be similar to the smallest space methods described below.
2. **Smallest space:** this set of closely related methods aim to find the optimal locations in Euclidean space such that the distances are as close as possible to the (inconsistent) network based distances. The two most common methods are:
   (a) **Force-directed:** A force is calculated with a positive component between connected nodes and a weaker negative component between all node pairs. The nodes are laid out randomly and all forces are calculated. The nodes are accordingly moved and these two steps are iterated until convergence. This is equivalent to an energy or stress minimisation of the graph as a system. The "forces" are often physics inspired. For example, in the Fruchterman-Reingold [47] and Kamada-Kawai [68] algorithms the positive force is analogous to a spring force and the negative force is analogous to the electrical force. The key difference is that the spring / attractive force is calculated between geodesic distance in Kamada-Kawai

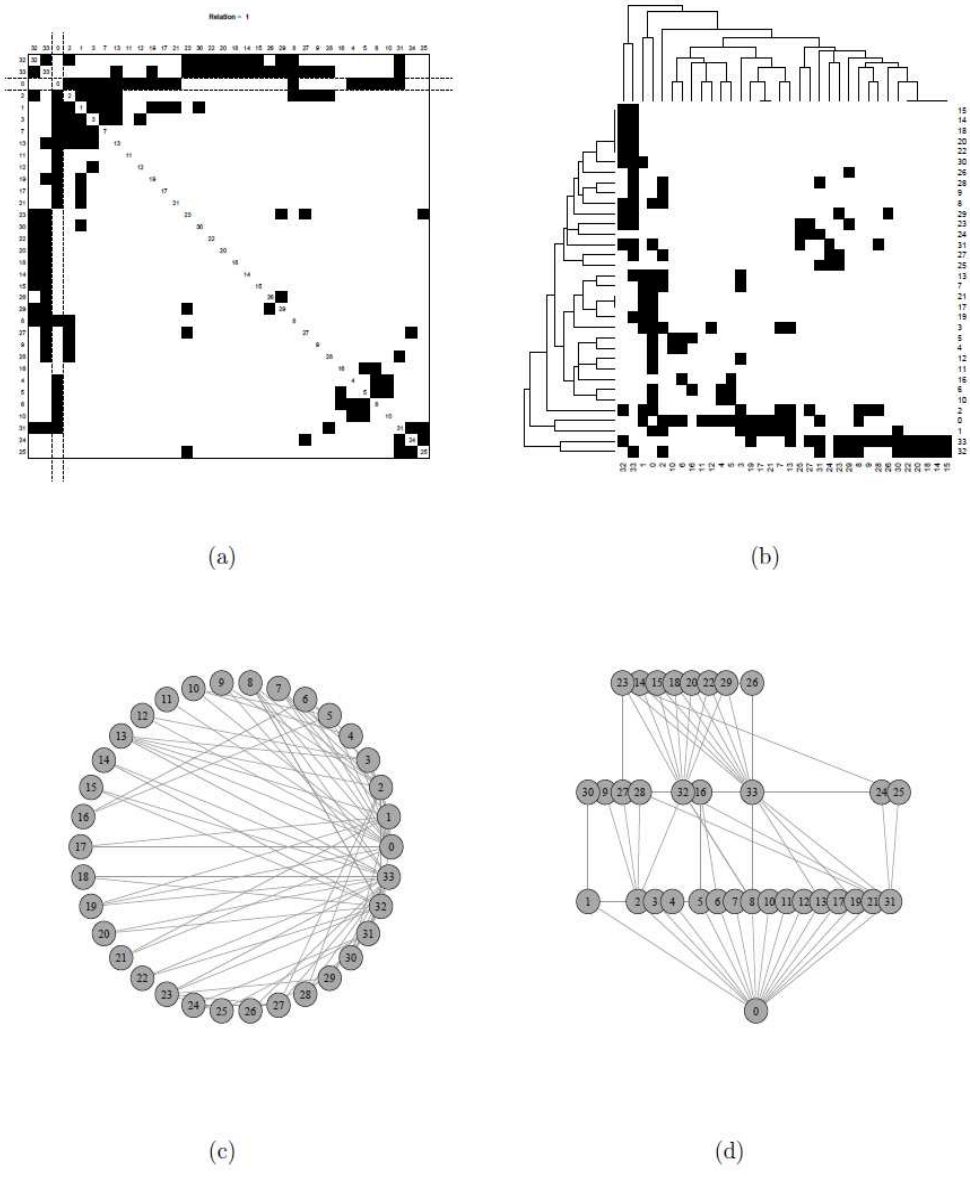


Figure 4: Four common visualisations for graphically displaying a network dataset. Zachary's Karate Club data is shown here, using the package `igraph` [34] for R. The blockmodel example makes use of the `sna` [21] package. (a) shows a plot of the adjacency matrix with rows and columns re-ordered to group equivalence classes. (b) shows a plot of the adjacency matrix with rows and columns re-ordered to group nodes close together in a hierarchical tree. The full dendrogram appears on the top and left. (c) shows a circle layout. (d) shows a tree layout using the Reingold-Tilford algorithm [91].

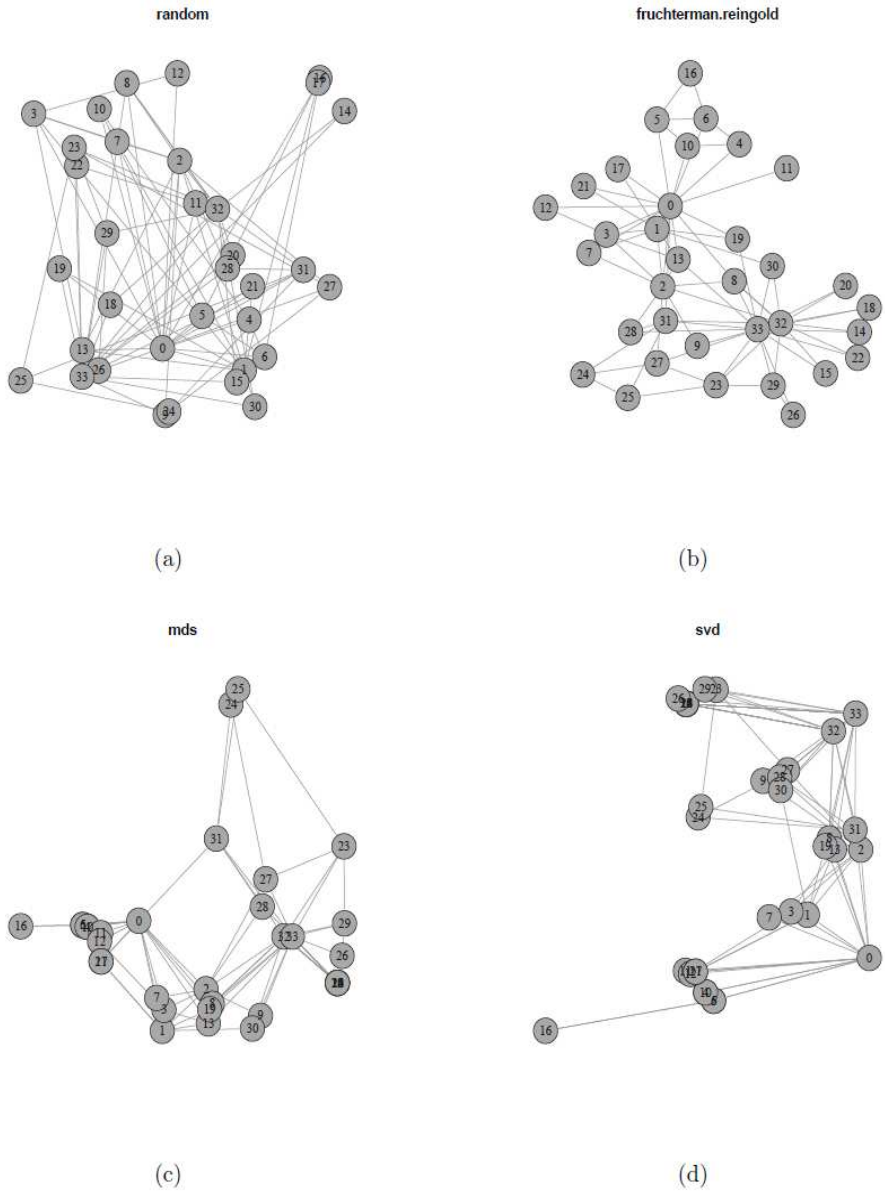


Figure 5: Four common layout methods for graphically displaying a network dataset. Zachary's Karate Club data is shown here, using the package `igraph` [34] for R. (a) shows a random layout. (b) shows a layout using the Fruchterman-Reingold algorithm [47]. (c) shows a layout based on multidimensional scaling. (d) shows a layout based on a singular-vector decomposition of the adjacency matrix.

and Euclidean distance in Fruchterman-Reingold. (Note that the plots in Figures 2 and 3 were generated using the Kamada-Kawai algorithm).

(b) **Multidimensional scaling:** The distance between two linked nodes is

taken to be zero and between unlinked nodes is one. Multidimensional scaling seeks to find the layout in a d-dimensional Euclidean space that maximises a similarity to this distance matrix. Alternatively, geodesic distances can be used in the multidimensional scaling algorithm.

3. **Spectral/eigenvalue decompositions:** A decomposition of the network based on eigenvectors of the adjacency matrix, or of the product of the adjacency matrix with its transpose, is taken. Coordinates based on the first two eigenvectors of this decomposition then provide the layout.
4. **Tree/hierarchical:** e.g. Reingold-Tilford [91]. The nodes are arranged in a tree-like structure with branches as links. Nodes at the same level are horizontally aligned.
5. **Shape based:** Nodes are arranged superimposed on a shape, for example a circle, star or sphere.

Finally, graph coarsening is a family of methods that aid the visualisation of large networks that are too complex to cleanly depict. A clustering of the graph (such as hierarchical) is performed based on the links. The graph of the clusters (at some chosen level) may be drawn using one of the above methods and graphs within clusters also drawn. Thickness of links in the upper graph(s) may be denoted using the number of links between the clusters. Interactivity plays a key role in this task as the user can effectively zoom in and out to view the visualisation of the graph at different granularities [39] provides a model for this process. There are other visualisation methods [e.g. 103], but the more popular methods tend to be examples or combinations of the above. [57] provides a survey of methods up to the year 2000.

3.3 Choosing a Layout Algorithm

Choosing a layout algorithm is usually a matter of trial and error, however knowledge of the characteristics of the network along with a pre-specified visualisation goal are useful in guiding this choice. There are also guiding principles to be found in [69] and [57]. These are problem specific, interactive and iterative. [112] provides a review of the current state-of-the-art in visualisation of large graphs.

3.4 Software

There are many software packages available for the visualisation of network data using the above methods. The R packages designed for network analysis such as `igraph` [34], `network` [22] and `RSiena` [111] contain visualisation tools using the common layout algorithms. `Visone` [9] includes an interface to R and `Pajek` [8] may be invoked from R. Other popular visualisation packages include `Tulip` [6], `Gephi`

[7] and Cytoscape [100]. It is worth noting that Cytoscape is extensively used in biological network visualisation applications.

4 Classical Models

These models assume a likelihood function for the network data given some underlying parameters. Estimates for these parameters are then inferred from the data. The models range in complexity from having a single scalar parameter for link probability to increasingly elaborate models constructed on counts of network summary statistics.

4.1 Erdős-Rényi

The Erdős-Rényi model [40, 41] is the most basic probabilistic model for network data. The model assumes that the presence and absence of edges between all pairs of nodes are i.i.d., where $y_{ij} = 1$ with probability θ and $y_{ij} = 0$ with probability $1 - \theta$. Hence, the probability of a particular network is given by

$$P(Y|\theta) = \prod_{i,j} \theta^{y_{ij}} (1-\theta)^{(1-y_{ij})},$$

where the product is over all pairs $i \neq j$ if the graph is directed and $i < j$ if the graph is undirected.

The Erdős-Rényi model has been studied extensively in the statistics and probability literature [see 38, for example]. In particular, the asymptotic properties of the model, as $N \to \infty$, have been studied in detail. The interplay of the values of p and N have a strong impact on the asymptotic model behaviour.

For example, suppose that $np \to \lambda$, then if $\lambda < 1$ the largest components will be of order $\log(N)$ in size, if $\lambda > 1$ a giant component of order N occurs and if $\lambda = 1$ the largest component will be of order $N^{2/3}$. Further, suppose that for some c, $np = \log(N) + c$. Then, the distribution of the number of isolated nodes converges to a Poisson distribution with parameter e^{-c} as $N \to \infty$.

However, due to the assumption of independent edges and equal probability of connectivity between pairs of nodes, the model is not appropriate for modelling many real world networks. Instead, it serves as a Null model, one in which there exists no structure.

This may be illustrated with a straightforward application to the Karate dataset. It is straightforward to estimate the maximium likelihood estimate $\hat{\theta} = 0.1375$. It then follows that the degree distribution should follow a Binomial$(33, 0.1375)$ distribution. Figure 6 shows the clear discrepancy between the observed and expected degree distribution for the data under the model. In particular, under this model the probability of any node having degree higher than 14 is only 5×10^{-5}, yet the

influential actors Mr. Hi and John A have degree of 16 and 17 respectively. It is also straightforward to construct a confidence interval for the expected number of triangles in the network. Again the network behaviour deviates from model assumptions: the observed number of triangles 43, is far greater than that in the 95% confidence interval $\{7.4, 23.4\}$.

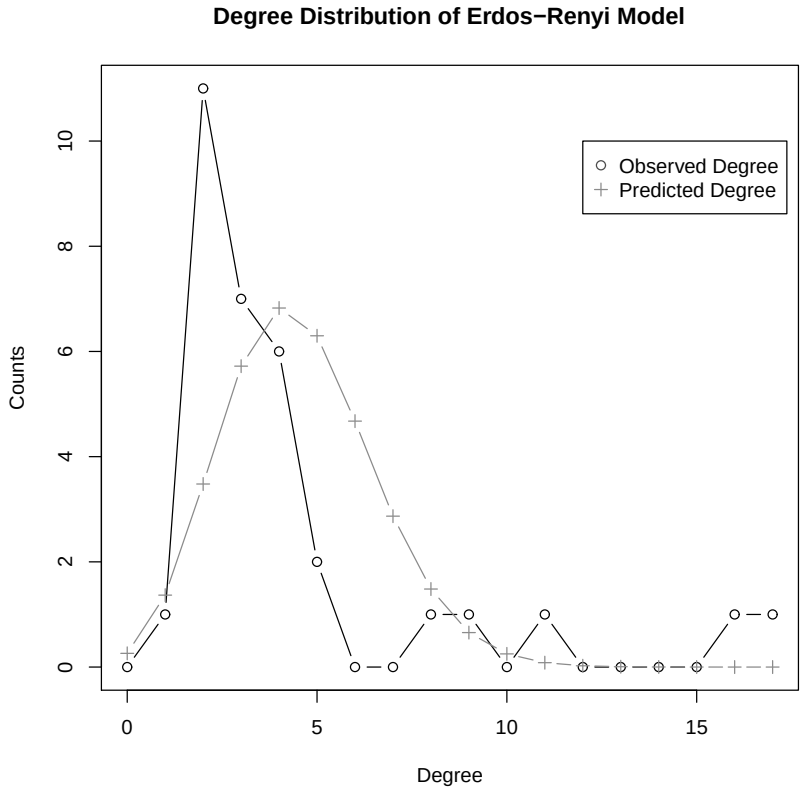


Figure 6: A line plot illustrating the differences between the observed degree distribution for the Karate dataset and its expected distribution under the Erdős-Rényi model.

4.2 p1 and p2

The p_1, p_2 and related models are essentially logistic regression models for the network dyads. They are a step up in complexity from the Erdős-Rényi model which models all ties as equally probable given a fixed probability of an individual tie for the network. The assumption of the independence of dyads in the p_1 and p_2 models implies that these models cannot capture common features of networks that involve more than two nodes, such as transitivity, clustering, etc.

4.2.1 The p1 model

This model for directed graphs was introduced by [62]. There are four possible states of linkage between two nodes n_i and n_j:

1. No link
2. n_i links to n_j only
3. n_j links to n_i only
4. n_i links to n_j and n_j links to n_i.

These four types of relationship are modelled respectively by the four terms below, with the probability of the entire network given by the exponent of the weighted sum:

$$P(Y = y) \propto \exp\left(\theta \sum_{i,j} y_{ij} + \sum_i \alpha_i \sum_j y_{ij} + \sum_j \beta_j \sum_i y_{ij} + \rho m\right), \tag{1}$$

where y is the observed adjacency matrix and $m = \sum_{i<j} y_{ij} y_{ji}$ is the number of mutual links.

The p_1 model thus contains three additional sets of parameters compared to the Erdős-Rényi model. Thus there is still the network-wide base rate of link probability, θ. The other parameters are for productivity α, attractiveness β and mutuality ρ. α and β are vectors with a separate value for each node.

When ρ and all α and β are zero, the model defaults to the Erdős-Rényi model. If ρ is zero the model is that link probability of a dyad is solely dependent on the degrees of the nodes involved. Allowing different values for ρ for each dyad leads to identifiability problems for the parameters. However, in the discussion of [62], [42] describe a model which "allows the effect of reciprocity to depend in a linear manner on the two actors in a dyad". This is achieved by having a dyad specific $\rho_{ij} = \rho + \rho_i + \rho_j$ and the ρ_i are normalised to sum to zero.

The full p_1 model may be expressed as a GLM as follows (reproduced from [110]):

$$\begin{aligned}
& P\left(y_{ij} = y_1, y_{ji} = y_2\right) \\
= \;& P\left(y_{ij} = y_1\right) P\left(y_{ji} = y_2 | y_{ij} = y_1\right) \\
= \;& \left(\exp\left(y_1(\theta + \alpha_i + \beta_j)\right) + \exp\left(y_1(\theta + \alpha_i + \beta_j + \rho) + \theta + \alpha_j + \beta_i\right)\right) / k_{ij}^{(1)} \\
\times \;& \exp\left(y_2(\theta + \alpha_j + \beta_i + y_1\rho)\right) / k_{ji}^{(2)},
\end{aligned} \tag{2}$$

where $k_{ij}^{(1)}$ and $k_{ji}^{(2)}$ are normalising constants with

$$\begin{aligned}
k_{ij}^{(1)} &= 1 + \exp(\theta + \alpha_i + \beta_j) + \exp(\theta + \alpha_j + \beta_i) \\
&+ \exp(2\theta + \alpha_i + \beta_j + \alpha_j + \beta_i + \rho) \qquad (3) \\
k_{ji}^{(2)} &= 1 + \exp(\theta + \alpha_j + \beta_i + y_1\rho). \qquad (4)
\end{aligned}$$

This formulation can be used for model fitting for the p_1 model as a Generalized Linear Model / Logistic Regression. According to [63], available software for fitting the p_1 model includes `UCINET` (sites.google.com/site/ucinetsoftware), `NetMiner`

(netminer.com) and `StOCNET` (gmw.rug.nl/ stocnet/StOCNET.htm). The `ergm` package for R [54] (see Section 4.4) may also be used to fit the p_1 model by setting the model as dependent on the edges, sender, receiver and mutual terms.

4.2.2 The p2 model

This is an extension of the p_1 model introduced by [110]. Essentially, the productivity α and attractiveness β are treated as random effects drawn from two underlying distributions. The p_2 model is thus a Generalized Linear Mixed Model. It was formulated with node specific attributes (covariates) in mind that play a part in determining productivity and attractiveness. These are modelled as fixed effects. If this covariate information is not available then the p_2 model is a more parsimonious version of p_1 where productivity and attractiveness are draws from two underlying distributions with mean zero and variances to be estimated from the data.

When covariate information on the nodes is available, two additional parameters per node attribute γ_1 and γ_2 must be estimated. These relate to the effects of the node attributes on productivity and attractiveness. `StOCNET` [63] may be used to fit the p_2 model. It can also be fit using R software for Generalized Linear Mixed Models such as `lme4` and `MCMCglmm`.

4.3 Block Models

Blockmodelling is a methodology which decomposes a network by mapping multiple nodes to one of a series of clusters (these are sometimes called positions or groups), $\mathcal{C} = \mathcal{C}_1, \ldots, \mathcal{C}_K$. The interaction between members of these clusters is described by the set of blocks $\mathcal{B} = \mathcal{B}_{11}, \mathcal{B}_{12}, \ldots, \mathcal{B}_{KK}$, where $\mathcal{B}_{ij}$ denotes the interaction between nodes in $\mathcal{C}_i$and $\mathcal{C}_j$. Nodes are mapped to the same cluster if they are considered to be equivalent with respect to some well defined specification. The network is then represented by the block interaction within and between clusters. Blockmodelling therefore provides a platform to describe a network at both a global and local level. An excellent overview of non-probabilistic blockmodelling methods and the many forms of equivalence in use is given in [26, Chapter 5].

Blockmodelling was first incorporated into a statistical framework with the introduction of the stochastic blockmodel [61]. This model sought to combine the probabilistic methodology of [62] with the blockmodelling approach first outlined in [79] by introducing the concept of *stochastic equivalence*. In effect this states that the conditional distribution of Y_{ij} depends only on the cluster membership of nodes n_i and n_j [116, Chapter 16].

Initial attempts to apply the stochastic blockmodel concentrated on the case where block membership is known a priori in the form of attribute data. [115] applied a particular version of this model by incorporating the stochastic blockmodel

into the p_1 model (Section 4.2). By introducing the indicator variable

$$d_{ijkl} = \begin{cases} 1 & \text{nodes } n_i \text{ and } n_j \text{ are in clusters } \mathcal{B}_k \text{ and } \mathcal{B}_l \\ 0 & \text{otherwise} \end{cases}$$

and introducing λ, a matrix of parameters of dimension $B \times B$ which describes blockmodel interaction, the p_1 model then becomes

$$\begin{aligned} P(Y = y) \quad &\propto \quad \exp(\rho m + \theta \sum_{i,j} y_{ij} + \sum_k \sum_l \lambda_{kl} \sum_i \sum_j y_{ij} d_{ijkl} \\ &+ \quad \sum_i \alpha_i \sum_j y_{ij} + \sum_j \beta_j \sum_i y_{ij}). \end{aligned} \tag{5}$$

In the more general case where attribute data is unknown, block membership assignation must be assigned with the use of latent variables [86]; see Section 5.2 for more details.

4.3.1 Software

The R package `sna` [21] may be used to fit stochastic blockmodels and thus estimate block structure in network datasets.

4.4 Exponential (family) Random Graph Models

Exponential (family) Random Graph Models (ERGMs) are a family of models that attempt to address the issues arising out of the assumption of dyadic independence in the p_1 and p_2 models. They are also referred to as p^* (p star) models and are an extension of the Markov Graphs [45] which adds triangle statistics to the p_1 model (Section 4.2). Subsequently, [117] generalised the model to include arbitrary network statistics. More recently, [75] propose a further extension of the ERGM model with the addition of a offset term to adjust for network size. [4] and [92] provide good introductions and [93] and [118] summarise recent developments in ERGM modelling.

As outlined in Section 4.2, p_1 and p_2 models cannot capture structure in networks pertaining to more than node pairs (e.g. transitivity). ERGMs do not assume dyadic independence and model the whole network as a single realisation arising from a distribution summarised by a collection of network sufficient statistics (Section 2.2). Specifically, the probability of the observed network y is in exponential family form which is proportional to the exponent of the sum of the network sufficient statistics times some unknown parameters:

$$P(Y = y) = \exp\left(\theta' S(y) - \gamma(\theta)\right) \tag{6}$$

where θ are the parameters of the model, $S(y)$ are the network sufficient statistics and $\gamma(\theta)$ is a normalising constant. This normalising constant is often very difficult to obtain as it involves summing over all possible networks with the observed

sufficient statistics, $S(y)$. This space contains $2^{N(N-1)/2}$ elements, so for a network of just 10 nodes there are 3.52×10^{13} possible network configurations. Therefore fitting ERGMs necessarily involves approximating this constant.

The network summary statistics are chosen by the analyst and may be calculated on higher than order two interactions. Typical choices include the number of edges, the number of triangles and the number of k-stars for various k values. [30] examine ERGMs with the degree sequence as a sufficient statistic.

Fitting the ERGM then involves finding estimates of the parameters for each of the network statistic terms in the model. There are several approaches to fitting ERGM models without recourse to summing over all possible networks; these include maximum pseudolikelihood estimation [107], Monte Carlo MLE [e.g. 109] and MCMC [e.g. 24]. More recently, [29] propose using a large deviations approximation to the normalising constant.

Maximum pseudolikelihood fitting involves computing change statistics for the network and finding a pseudo MLE for the parameters using a pseudolikelihood approximation of the likelihood. The pseudolikelihood approximation [11] which is the product of the full conditionals for each dyad given the rest of the observed network is computed; the log-odds of a particular dyad being linked is θ times the change in the observed network sufficient statistics in switching from that dyad being linked to unlinked.

Following [92], denoting the change in network statistics when Y_{ij} is switched from 1 to 0 as d_{ij}, the log-odds of Y_{ij} being one, given the rest of the network is

$$\log\left(\frac{Pr(Y_{ij}=1|y_{ij}^c)}{Pr(Y_{ij}=0|y_{ij}^c)}\right) = \theta' d_{ij}(y) \tag{7}$$

where y_{ij}^c) is all entries of the adjacency matrix except ij. Note that only the change statistics that involve Y_{ij} need be included in the calculation. Estimation of the parameters θ then proceeds via maximum likelihood estimation in a manner similar to logistic regression.

However, the properties of the approximation are not well known. Measures of model fit are problematic as networks cannot be easily simulated from the fitted model. Furthermore, logistic regression assumes independent observations; this assumption is clearly wrong for network data, potentially leading to biased parameter estimates and standard errors which may be too small [93, 92].

For these reasons, Monte-Carlo based inference is preferred, where possible. Note that this does not imply that a Bayesian perspective is necessarily taken, although Bayesian estimation of ERGMs has recently been addressed in [24]. Frequentist MCMC methods for ERMGs (MCMCMLE) refine approximate parameter estimates by comparing the observed network against a set of simulated networks given a parameter configuration [104, 65]. Essentially, a computationally feasible sample of

networks consistent with the observed summary statistics is used in lieu of the set of all consistent networks.

Starting with an arbitrary estimate $\theta^{(0)}$, MCMC is used to sample M networks from an ERGM with these parameters. Denoting these sampled networks via their adjacency matrices $Y^{(1)}, \ldots, Y^{(M)}$ then the MCMC log-likelihood is given by

$$\begin{aligned} \log P_M(Y|\theta) &= \theta' S(Y) - \log(\gamma(\theta^{(0)})) \\ &- \log\left(\frac{1}{M}\sum_{m=1}^{M} \exp\left(\theta' S(Y^{(m)}) - \theta'^{(0)} S(Y^{(m)})\right)\right) \end{aligned} \tag{8}$$

The limit as $M \to \infty$ of this MCMC log-likelihood is equal to the log-likelihood of the ERGM. The argmax of θ is referred to as the MCMCMLE of the network. The performance depends on the choice of $\theta^{(0)}$; a poor choice leads to convergence to a local maximum. [48] suggest an iterative procedure: start with the maximum pseudolikelihood solution for $\theta^{(0)}$ as the initial choice then calculate the MCMCMLE; now use this as $\theta^{(0)}$ to obtain a new MCMCMLE and iterate until convergence (i.e. change in the MCMCMLE θ is below a specified threshold).

The other issue is in sampling networks from an ERGM with specified parameters. A Gibbs sampler is developed in [104] and this work revealed some shortcomings of the ERGM specification. In both inferential approaches, model degeneracy may be an issue. Simply put, degeneracy refers to the situation in which only a few networks have appreciable probability given the model. These few networks may include the full or empty networks and will not be useful in practice. If degeneracy or *near degeneracy* occur then the parameter estimates may not converge and maximum pseudolikelihood estimators will yield misleading results in these cases. In these cases, the model is poorly specified and no estimation procedure will fix this (see [55] for further details on degeneracy in statistical models of social networks). Networks simulated from the fitted model "bear little resemblance at all to the observed network".

Therefore new specifications of ERGM models have emerged with network summary statistics specifically chosen to address the degeneracy problems. These include conditioning on the number of edges [104] or the inclusion of alternating k-stars and alternating k-triangles [93]. [64] note that "geometrically weighted degree, edgewise shared partner, and dyad-wise shared partner statistics [are] equivalent to the alternating k-star, k-triangle, and k-twopath statistics, respectively". Alternating refers to setting successive signs of the network statistic to be plus and minus when indexing over the order of the statistic. For example, if 2-triangle counts have a positive sign then 3-triangle counts are given a negative sign, etc.

[64] discuss goodness of fit for ERGM models. They propose comparison of observed network statistics with the sample distributions of other statistics calculated on networks simulated from the fitted model. They examine the degree distribu-

tion(s), edgewise shared partners distribution and the geodesic distance distribution. Assessment is performed visually using boxplots summarising simulated values with observed values overlaid or via a performance metric based on these (such as p-values for the observed statistic under the empirical distribution of statistics).

Extensions of ERGMs to dynamic network datasets include [56].

4.4.1 Software

The `ergm` package [54] which is part of the `statnet` [53] suite of packages provides comprehensive toolsets for the analysis of network data using ERGMs. The add on `Bergm` package [23] contains a collection of functions implementing Bayesian analysis for ERGMs using Markov Chain Monte Carlo. SIENA (available as standalone package or as an R package `RSiena` [111]) also fits ERGM models.

5 Latent Variable Models

These models seek to explain the structure exhibited by network data via an additional layer of modelling. A broad class of model can be used in such a setting - as [60] notes, any statistical model for a network in which the nodes are exchangeable may be expressed as a latent variable model. The network data is modelled as dependent on a latent or unobservable set of random variables, which are in turn subject to some modelling assumptions that impose structure. The nested, hierarchical structure of such models means that they can sometimes be expressed as a probabilistic graphical model [67, 66, 114, 14]. Inference is then performed, usually in a Bayesian framework, to obtain parameter estimates or posterior densities given the observed network.

5.1 Latent Space Models

Latent space models were introduced by [58] under the basic assumption that each node n_i has an unknown position z_i in a d-dimensional Euclidean latent space. Network edges are assumed to be conditionally independent given the latent positions, and the probability of an edge (η_{ij}) between nodes n_i and n_j is modelled as a function of their positions. Generally, in these models the smaller the distance between two nodes in the latent space, the greater their probability of being connected. An important feature of these models is that they easily and naturally account for *reciprocity* and *transitivity* (Section 2.1).

In the case where additional edge covariate information x_{ij} is observed, these models can account for *homophily* by attributes. Node covariate information is typically converted to edge covariates using sums or differences (either directed or

absolute); thus an edge is more (or less) likely to occur between actors that have similar attributes than between those who do not.

$$P(Y|Z, X, \theta) = \prod_{i \neq j} P(y_{ij}|z_i, z_j, x_{ij}, \theta). \tag{9}$$

The Distance Model and the Projection Model

[58] proposed two main latent space models; the distance model and the projection model. The former is the most widely used for its simple interpretation, since it depends directly on the distance between the actors in the social space. The most used distance is the Euclidean distance, but any distance $d_{ij} = d(z_i, z_j)$ satisfying the triangle inequality $d_{ij} \leq d_{ik} + d_{kj}$ for all $\{i, k, j\}$ triples may be used. This model supposes the network to be intrinsically symmetric since it has the feature of being *reciprocal*; if $y_{ij} = 1$ then the probability of $y_{ji} = 1$ is large. So the distance model is particularly suitable for undirected graphs or directed graphs that exhibit strong reciprocity. The distance model is:

$$\eta_{ij} = \log \text{odds}(y_{ij} = 1|z_i, z_j, x_{ij}, \alpha, \beta) = \alpha + \beta' x_{ij} - |z_i - z_j|. \tag{10}$$

The projection model is more adequate for strongly asymmetric graphs since it is founded on the assumption that the probability of observing an edge between two actors i and j depends on the angle that they create in the Bilinear latent space; if the angle is small the probability of having an edge is large, and if the angle is obtuse the probability of having an edge is small. So the projection model is:

$$\eta_{ij} = \log \text{odds}(y_{ij} = 1|z_i, z_j, x_{ij}, \alpha, \beta) = \alpha + \beta' x_{ij} - \frac{|z_i' z_j|}{|z_j|}. \tag{11}$$

The Latent Position Cluster Model

[52] proposed the Latent Position Cluster Model (LPCM), a new model which extends the latent space distance models to allow for model based clustering of the nodes. A spherical Gaussian mixture model structure is assumed for the latent positions:

$$z_i \sim \sum_{g=1}^{G} \lambda_g \text{MVN}_d(\mu_g, \sigma_g^2 \mathbf{I}) \tag{12}$$

where λ_g is the probability that a node belongs to the gth group, and $\sum_{g=1}^{G} \lambda_g = 1$; this structure allows for clusters of highly connected nodes.

The Sender and Receiver Random Effects

[59] proposed to take into consideration the *degree heterogeneity*; the tendency of some actors to send and/or receive edges more than others. This method allows the modelling of asymmetric networks within the distance model. [76] proposed a model that explicitly considers all the four features noted above. In undirected graphs there is only one parameter, δ_i, called the sociality factor. This denotes the propensity of each actor n_i to form edges with other actors.

$$\eta_{ij} = \log \text{odds}(y_{ij} = 1 | z_i, z_j, x_{ij}, \alpha, \beta) = \alpha + \beta' x_{ij} - |z_i - z_j| + \delta_i + \delta_j. \tag{13}$$

In directed graphs the sociality effect in the dyad y_{ij} depends on two parameters: the sender random effect δ_i and the receiver random effect γ_j.

$$\eta_{ij} = \log \text{odds}(y_{ij} = 1 | z_i, z_j, x_{ij}, \alpha, \beta) = \alpha + \beta' x_{ij} - |z_i - z_j| + \delta_i + \gamma_j \tag{14}$$

where $\delta_i \sim \mathcal{N}(0, \sigma_\delta^2)$ and $\gamma_i \sim \mathcal{N}(0, \sigma_\gamma^2)$, and the variances σ_δ^2 and σ_γ^2 measure the heterogeneity in the propensity to send and receive edges. For undirected networks, there is a single sociality effect for each node. A fit of this model with two groups to the Karate club data is provided in Figure 7. The model correctly identifies the two factions to which each actor belongs and the social random effects are higher for the more central nodes, with Mr. Hi and John A having the highest values and therefore the largest plotting symbols (pie charts) in Figure 7.

The Mixture of Experts Latent Position Cluster Model

[51] proposed the mixture of experts latent position cluster model to extend the latent position cluster model within a mixture of experts framework, assuming that the mixing proportions $(\lambda_1, \ldots, \lambda_G)$ are node specific and can be modelled as a Multinomial logistic function of their covariates $\mathbf{w}_i^T = (w_{i1}, \ldots, w_{ip})$ where the probability of belonging to each of $G-1$ clusters are compared to a baseline cluster, usually $g = 1$. The distribution of z_i is assumed to be:

$$z_i \sim \sum_{g=1}^{G} \lambda_g(\mathbf{w}_i) \text{MVN}_d(\mu_g, \sigma_g^2 \mathbf{I}) \tag{15}$$

where

$$\lambda_g(\mathbf{w}_i) = \frac{\exp(\tau_{g0} + \tau_{g1} w_{i1} + \ldots + \tau_{gp} w_{ip})}{\sum_{g'=1}^{G} \exp(\tau_{g'0} + \tau_{g'1} w_{i1} + \ldots + \tau_{g'p} w_{ip})} \tag{16}$$

$(\tau_{10}, \ldots, \tau_{1p}) = (0, 0, \ldots, 0)$ and $\sum_{g=1}^{G} \lambda_g(\mathbf{w}_i) = 1$.

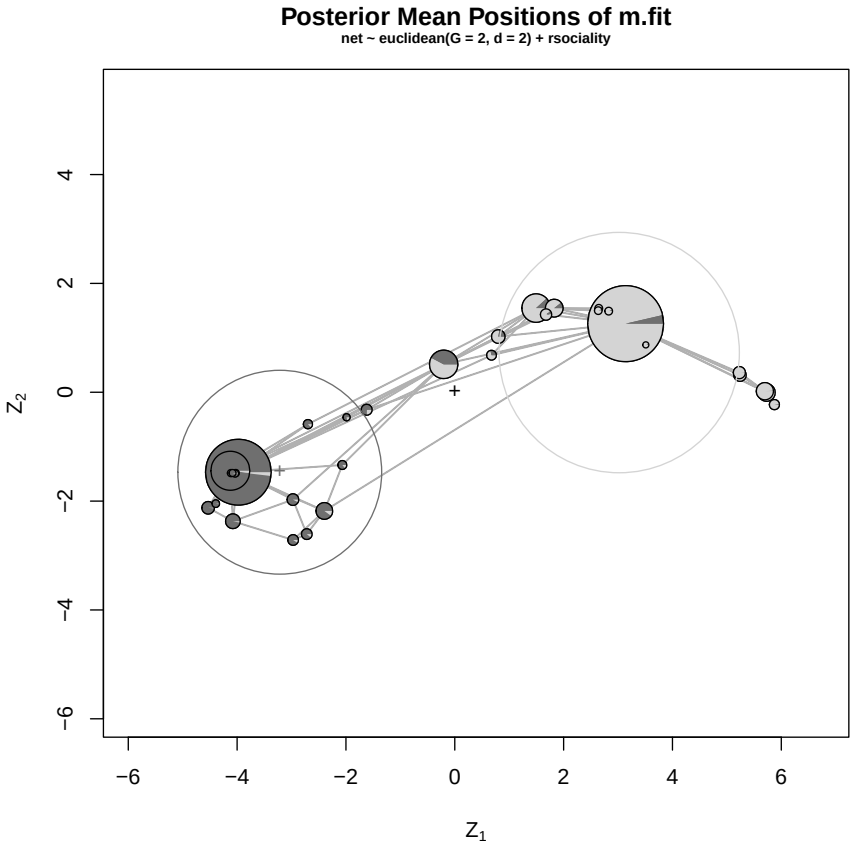


Figure 7: Plot of the posterior latent positions for two groups under the Latent Position Cluster Model for Zachary's Karate Club dataset. Random sender effects are depicted via the size of the nodes and the pie charts illustrate the posterior probability of belonging to each of the two groups. The R package `latentnet` [74] was used to perform the inference and create the image.

5.1.1 Estimation

For all variants of the latent space models presented above the log-likelihood is of the form:

$$\text{loglikelihood} = \log P(Y|\eta) = \sum_{i \neq j} \{\eta_{ij} y_{ij} - \log(1 + \exp(\eta_{ij})\} \tag{17}$$

To estimate the model the main approaches suggested are maximum likelihood estimation and a fully Bayesian approach that involves MCMC sampling.

The maximum likelihood approach is a fast method that provides points estimates of the distances between the nodes since the log-likelihood is a convex function of the distances. The drawback of this approach is that the latent positions should be approximated using multidimensional scaling of the distances since the log-likelihood is not generally a convex function of the latent positions. The set of latent positions found with this approach provides good starting values for non-

convex optimisation methods. Another drawback is that the estimation is done in two steps; in the first step the maximum likelihood estimates of the latent space model is computed considering no clusters, and in the second step the maximum likelihood estimates of the mixture model conditioned on the latent positions estimated at the previous step are calculated.

A fully Bayesian approach allows the estimation of all the parameters and the latent position simultaneously e.g. via MCMC sampling. This approach usually gives better results than the two stage MLE, but it is more computationally and algebraically intensive.

Similarly to [2] who use a variational approximation to fit a mixed-membership stochastic blockmodel, [97] propose a variational Bayesian inference routine to approximate the posterior distribution of the parameters in the LPCM. More recently, [90] propose a likelihood approximation using case-control sampling to improve the efficiency of fitting the LPCM model.

5.1.2 Software

The R package `latentnet` [74, 73] provides two-stage maximum likelihood estimation, MCMC based inference and minimum Kullback-Leibler [102] estimation for the LPCM for both Euclidean and Bilinear latent spaces. `gbme` [59] is R code which uses an inner kernel product that is similar to the projection model, but does not provide modelling or estimation of clusters. `VBLPCM` [96] is an R package that performs fast variational Bayesian inference of the LPCM for Euclidean latent spaces.

5.2 Latent Block Models

5.2.1 Stochastic Block Model

Blockmodels were discussed in Section 4.3 and they are latent variable models in the case where the cluster memberships are unknown. The stochastic blockmodel as formulated by [105] and [86] introduced cluster membership as a latent variable where $\mathbb{P}(n_i \in \mathcal{C}_k) = \theta_k$. Let A denote the cluster membership of each node, where $A_i = k$ if node n_i belongs to cluster $\mathcal{C}_k$. The dyadic state Y_{ij} is modelled as $\mathbb{P}(y_{ij} \mid A) = \eta(A_i, A_j)$, where η is the model for interactions; the most common model assumes a $K \times K$ matrix of probabilities, denoted as B, and $\eta(A_i, A_j)$ is a Bernoulli model with probability $b_{A_i A_j}$. Inference for this model may be performed via Gibbs sampling.

Recent studies have considered the stochastic blockmodel within a broader context. [12] studied the connections between modularity (Section 2.4) and stochastic block models, while [13] discusses some of the asymptotic properties of a class of models of which stochastic blockmodels are a subset. [94] and [31] also consider

the asymptotic performance of the method with respect to the misclassification of nodes, [94] comparing the method (favourably) to spectral clustering (Section 2.4), [31] within a maximum likelihood estimation framework.

5.2.2 Mixed Membership Stochastic Block Model

The Mixed Membership Stochastic Blockmodel (MMSB) [2] combines the approach of [86] with the latent Dirichlet allocation model of [15]. In this framework, as in in Sections 4.3 and 5.2.1, a directed graph Y is generated by K clusters, $\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_K\}$, with any two nodes in the same cluster being considered stochastically equivalent.

However, in contrast to other latent class models in Sections 4.3, 5.1 and 5.2.1, the position of a node i may change depending on which node j in the network it is interacting with. Each node may therefore have multiple cluster memberships within a network; node i has an individual probability π_{ik} of belonging to cluster k while interacting with another node in the network. Indicator variables $\overrightarrow{z}_{i\rightarrow j}$ and $\overrightarrow{z}_{i\leftarrow j}$ then denote sender and receiver cluster membership for each interaction between nodes i and j respectively. Once cluster membership is accounted for, nodal interaction y_{ij} is modelled, as in Section 4.1, as a Bernoulli with a $K \times K$ probability matrix B of inter and intra block interactions.

Networks are thus assumed to be drawn from the following data generation procedure:

- For each node $n_i \in \mathcal{N}$:
  - Draw a K dimensional mixed membership vector $\overrightarrow{\pi}_i \sim \text{Dirichlet}(\overrightarrow{\alpha})$
- For each pair of nodes $(n_i, n_j) \in \mathcal{N} \times \mathcal{N}$:
  - Draw membership indicator for the initiator, $\overrightarrow{z}_{i\rightarrow j} \sim \text{Multinomial}(\overrightarrow{\pi}_i)$
  - Draw membership indicator for the receiver, $\overrightarrow{z}_{i\leftarrow j} \sim \text{Multinomial}(\overrightarrow{\pi}_j)$
  - Sample the value of their interaction, $y_{ij} \sim \text{Bernoulli}(\overrightarrow{z}_{i\rightarrow j} B \overrightarrow{z}_{i\leftarrow j})$

A number of extensions and alternatives to the MMSB model have recently been developed. [121] develop a dynamic mixed membership stochastic blockmodel (dMMSB). [77] and [82] develop overlapping stochastic block models that allow nodes to have full membership of more than one block, thus providing an alternative version of mixed membership.

5.2.3 Application to Karate Dataset

While [2] implement a variational approximation to the model in order to perform inference on a network, to our knowledge no software is as yet publicly available. Conversely, a collapsed Gibbs sampler is available as part of the `R` package `lda` [27], with no supporting documentation other than the user's manual. Since collapsed

Gibbs samplers have recently been successfully applied to [88] we believe that it is implemented in a similar manner.

A collapsed Gibbs sampler run of 100,000 iterations was performed on the karate data, with two blockmodels assumed to underpin the model as chosen by BIC. Uniform priors were set for all parameters. The results from the sampler provide a nice illustration of the differences between blockmodel estimates and other methods, such as latent-space models, which explicitly attempt to cluster nodes. The estimate for the interaction matrix B takes the values

$$\hat{B} = \begin{pmatrix} 0.00 & 0.45 \\ 0.44 & 0.30 \end{pmatrix}.$$

Note that the probability of interaction is higher on the off-diagonals, that is between the two blocks. This is because blockmodels group nodes whose behaviour is similar, rather than explicitly partitioning highly connected nodes. We can also note that the second block is a great deal smaller than the first. We can therefore think of the second block as having strong influence over the first block, who do not link with each other. In particular, note that nodes 0, 32 and 33 have strong probability of membership to the second group, which corresponds with their strong centrality scores from Section 2.

We also provide a plot of the fit when three blocks are modelled in Figure 8, with three chosen here to highlight the compositional property of the blockmodel memberships as a 3-composition simplex reduces to a two-dimensional plot. In this case

$$\hat{B} = \begin{pmatrix} 0.70 & 0.00 & 0.00 \\ 0.00 & 0.02 & 0.90 \\ 0.06 & 0.78 & 0.75 \end{pmatrix},$$

where bottom left corresponds to the first block, bottom right is the second and the top is the third.

5.2.4 Software

Algorithms for Variational Bayesian inference under the MMSB are detailed in [2]. A collapsed Gibbs sampler of the model is available as part of the `R` package `lda` [27].

6 Goodness-of-Fit and Validation

There is much scope for further work on the assessment of goodness-of-fit and model choice in the network analysis setting. Quantitative methods of model assessment currently fall into four overlapping categories; (1) comparison with ground truth / nodal attributes, (2) link prediction, (3) goodness-of-fit diagnostics and (4) model

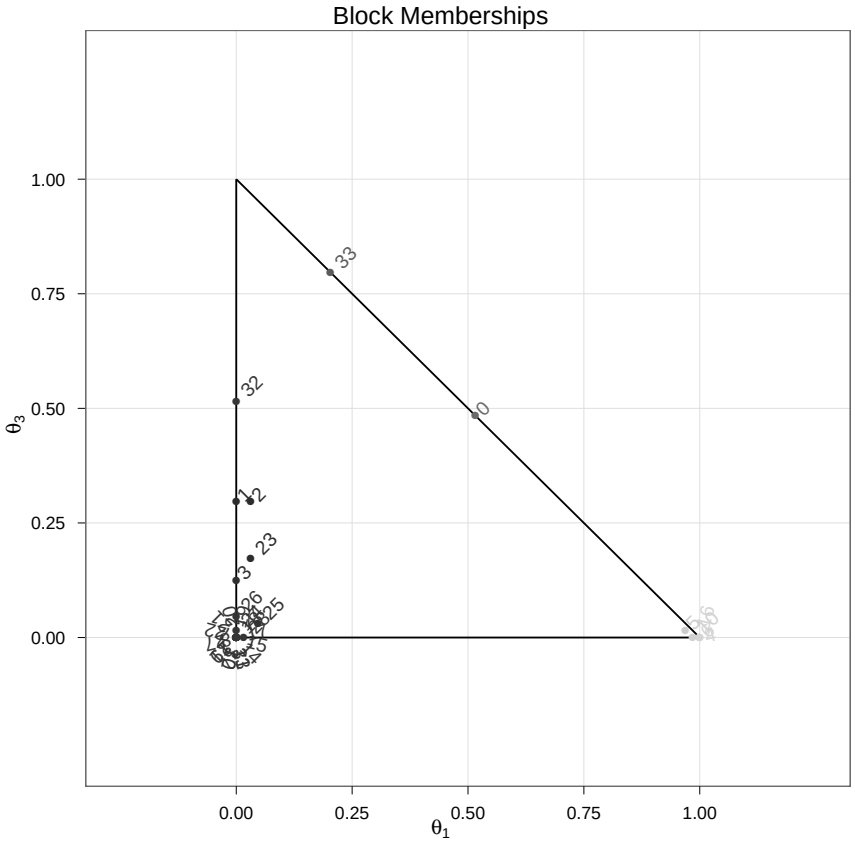


Figure 8: Plot of the posterior latent groupings under the MMSBM for Zachary's Karate Club dataset. The R package `lda` [27] was used to produce the plot.

comparison via e.g. information criteria. We provide only a brief summary of these methods here.

Comparison with ground truth involves comparing aspects of the fitted model to observed nodal attributes. For example, in the Karate club dataset it is known to which two clubs the actors split; thus any community detection algorithm that separates the nodes into clusters that are consistent with this split is deemed to be a good model. Such comparison is valid only when the ground truth is not incorporated into the model and can only be used to validate a model on a problem for which the community structure is known. Of course, it is unusual to have this luxury in practice and one of the following methods should be used in applied network analysis. Comparison with a ground truth is more often used to validate a method of model comparison.

Link prediction involves comparison of the predictive probabilities under the fitted model for all possible links with the observed links / non-links. This method is used for generative models and inspection of the fit is often performed visually. For example, two boxplots of the probabilities of a link are created; one for the non-links and one for the links. If the probability of a link is systematically lower

for the non-links than the links then the model is deemed to be a good fit to the observed network. In the case of the latent space model, [58] define a network as d_k *representable* if there exists a set of nodal positions in k dimensional Euclidean space such that the distances between only the linked nodes are all less than a threshold distance. Conversely, only all non-linked nodes are separated by a distance greater than this threshold. For such cases, the boxplots described above will be entirely non-overlapping. Such analysis is applicable to any probabilistic model where the likelihood of individual links is computable.

Goodness-of-fit diagnostics as detailed in [64] represent perhaps the most sophisticated and developed methods for model validation. This method was developed specifically for the ERGM (Section 4.4) class of models but has also been applied to the latent space models [73]. Indeed, the method is applicable to any fitted model from which networks are readily simulated. The method is motivated by the observation that "when ERGM parameters are estimated and a large number of networks are simulated from the resulting model, these networks frequently bear little resemblance at all to the observed network" [55]. The method involves simulation of networks from the fitted model; summary statistics (such as degree distribution, shared partners distribution, geodesic distance, etc) derived from these simulated networks are then compared with the corresponding observed values. [64] advocate using such higher order statistics that are not functions of the ERGM parameters to "provide a strong independent criterion for goodness of fit". Comparison is first performed visually; for example the observed degree distribution is overlain on the boxplots of the degree distributions of the simulated networks.

Model comparison may involve either comparison of competing models under an information criterion such as BIC or assessment of both models' goodness-of-fit as above. BIC is used in [52] to choose the number of clusters or communities present in a network fitted using the Latent Position Cluster Model (Section 5.1). [64] perform a comparison of parameter selection using AIC to selection based on statistics derived from the goodness-of-fit diagnostics above with consistent results.

7 Conclusion

We have presented a concise summary of a number models, methods and software for the statistical analysis of network datasets. We explored classical models in which the presence of a link between two nodes depends on the network graph structure and latent variable models in which the presence of a link in the network depends on the presence of a latent variable. We have considered only binary static networks as these are the most analysed family of networks in the literature. Our inclusion of a recurring example [122] and reference to software makes this paper

a useful tutorial and pathway into the subject of statistical network analysis. We suggest software (mainly in the form of R packages) to apply most of the methods and models described in the paper.

In recent times, network data has become pervasive. Modern data collection methods can allow for network data to be collected over time with greater ease than previously. When analysing temporal network data it is common to aggregate across time-points or to consider single time snapshots of the network, and to analyse these data using static network techniques. However, proper dynamic methods for social network analysis is a rapidly growing area. Current methodology in the statistical literature typically involves adding a smoothed-over-time component to a model for static network data [e.g. 121]. There is much scope for modelling innovations for such temporal data.

We have considered only binary networks; i.e. an edge either exists or does not but takes no other values. Other network types in the literature include weighted edges and these are typically modelled in similar fashion to binary networks but replacing the logistic part of the model with e.g. a Poisson likelihood. However, other parts of the model may require alteration; for example when fitting ERGMs the binary link based network summary statistics are no longer appropriate and need to be extended to more general situations [see 72].

Many challenges remain in the field of statistical analysis of network data. Perhaps the greatest challenge is scaling current methodology to huge graphs. The internet provides a wealth of network datasets, with the world-wide-web itself forming perhaps the largest. Most (or all) statistical methods in this paper may be cast as modelling the links and non-links as samples from a stochastic process driven by some underlying structure. Therefore, these methods necessarily scale as $\mathcal{O}(N^2)$ where N is the number of nodes. This makes extension of such methods to networks with more than a few thousand of nodes impracticable. In contrast, many descriptive and algorithmic approaches to analysing network data exploit network sparsity. However, [28] does provide a model-based analysis that exploits sparsity in a mixed membership stochastic blockmodelling setting. [90] use a stratified case-control sampler to reduce the computational complexity of the likelihood evaluation to $\mathcal{O}(N)$. This approximation can in principle be applied to any statistical method based on computing the likelihood of all possible links in the network.

References

[1] B. Adamcsek, G. Palla, I. J. Farkas, I Derényi, and T. Vicsek. CFinder: locating cliques and overlapping modules in biological networks. *Bioinformatics*, 22(8):1021–1023, 2006.

[2] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed-membership stochastic blockmodels. *Journal of Machine Learning Research*, 9:1981–2014, 2008.

[3] Edoardo M. Airoldi, David M. Blei, Stephen E. Fienberg, Anna Goldberg, Eric P. Xing, and Alice X. Zheng. *Statistical Network Analysis: Models, Issues and New Directions*, volume 4503 of *Lecture Notes in Computer Science*. Springer, Berlin, 2007.

[4] Carolyn J Anderson, Stanley Wasserman, and Bradley Crouch. A p* primer: logit models for social networks. *Social Networks*, 21(1):37 – 66, 1999.

[5] P. Arabie and L. J. Hubert. An overview of combinatorial data analysis. In P. Arabie, L. J. Hubert, and G. De Soete, editors, *Clustering and Classification*, pages 5–63. World Scientific, River Edge, NJ, 1996.

[6] D. Auber. Tulip : A huge graph visualisation framework. In P. Mutzel and M. Jünger, editors, *Graph Drawing Softwares*, Mathematics and Visualization, pages 105–126. Springer-Verlag, 2003.

[7] Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. Gephi: An open source software for exploring and manipulating networks. In Eytan Adar, Matthew Hurst, Tim Finin, Natalie S. Glance, Nicolas Nicolov, and Belle L. Tseng, editors, *Proceedings of the Third International Conference on Weblogs and Social Media (ICWSM 2009)*. The AAAI Press, 2009.

[8] Vladimir Batagelj and Andrej Mrvar. Pajek analysis and visualization of large networks. In Petra Mutzel, Michael Jünger, and Sebastian Leipert, editors, *Graph Drawing*, volume 2265 of *Lecture Notes in Computer Science*, pages 8–11. Springer, Berlin/Heidelberg, 2002.

[9] Michael Baur, Marc Benkert, Ulrik Brandes, Sabine Cornelsen, Marco Gaertler, Boris Köpf, Jürgen Lerner, and Dorothea Wagner. Software for visual social network analysis. In Petra Mutzel, Michael Jünger, and Sebastian Leipert, editors, *Graph Drawing*, volume 2265 of *Lecture Notes in Computer Science*, pages 554–557. Springer, Berlin/Heidelberg, 2002.

[10] Pavel Berkhin. A survey on PageRank computing. *Internet Mathematics*, 2:73–120, 2005.

[11] Julian Besag. Statistical analysis of non-lattice data. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 24(3):179–195, 1975.

[12] Peter J. Bickel and Aiyou Chen. A nonparametric view of network models and Newman-Girvan and other modularities. *Proceedings of the National Academy of Sciences*, 106(50):21068–21073, 2009.

[13] Peter J. Bickel, Aiyou Chen, and Elizaveta Levina. The method of moments and degree distributions for network models. *Annals of Statistics*, 39(5):2280–2301, 2011.

[14] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[15] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[16] Phillip Bonacich. Power and centrality: A family of measures. *American Journal of Sociology*, 92(5):1170–1182, 1987.

[17] Phillip Bonacich. Some unique properties of eigenvector centrality. *Social Networks*, 29(4):555–564, 2007.

[18] Ulrik Brandes, Daniel Delling, Marco Gaertler, Robert Görke, Martin Hoefer, Zoran Nikoloski, and Dorothea Wagner. On finding graph clusterings with maximum modularity. In Andreas Brandstädt, Dieter Kratsch, and Haiko Müller, editors, *Graph-Theoretic Concepts in Computer Science*, volume 4769 of *Lecture Notes in Computer Science*, pages 121–132, 2007.

[19] 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(2):172–188, 2008.

[20] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[21] Carter T. Butts. *sna: Tools for Social Network Analysis*. University of California, Irvine, 2010. R package version 2.1-0.

[22] Carter T. Butts, Mark S. Handcock, and David R. Hunter. *network: Classes for Relational Data*. Irvine, CA, 2011. R package version 1.7.

[23] Alberto Caimo and Nial Friel. *Bergm: Bayesian inference for exponential random graph models*, 2010. R package version 1.4.

[24] Alberto Caimo and Nial Friel. Bayesian inference for exponential random graph models. *Social Networks*, 33(1):41 – 55, 2011.

[25] Jérôme Callut, Kevin Françoisse, Marco Saerens, and Pierre Dupont. Semi-supervised classification from discriminative random walks. In *Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases - Part I*, ECML PKDD '08, pages 162–177, Berlin/Heidelberg, 2008. Springer-Verlag.

[26] Peter J. Carrington, John Scott, and Stanley Wasserman. *Models and methods in social network analysis*. Cambridge University Press, 2005.

[27] Jonathan Chang. *lda: Collapsed Gibbs sampling methods for topic models.*, 2010. R package version 1.2.1.

[28] Jonathan Chang and David M. Blei. Hierarchical relational models for document networks. *Annals of Applied Statistics*, 4(1):124–150, 2010.

[29] Sourav Chatterjee and Persi Diaconis. Estimating and understanding exponential random graph models. Technical report, Stanford University, 2012. Available at arXiv.org:1102.2650.

[30] Sourav Chatterjee, Persi Diaconis, and Allan Sly. Random graphs with a given degree sequence. *Annals of Applied Probability*, 21(4):1400–1435, 2011.

[31] D. S. Choi, P. J. Wolfe, and E. M. Airoldi. Stochastic blockmodels with growing number of classes. *Biometrika*, To appear, 2011.

[32] N. A. Christakis and J. H. Fowler. The collective dynamics of smoking in a large social network. *New England Journal of Medicine*, 358(21):2249–2258, 2008.

[33] Nicholas A. Christakis and James H. Fowler. The spread of obesity in a large social network over 32 years. *New England Journal of Medicine*, 357(4):370–379, 2007.

[34] Gabor Csardi and Tamas Nepusz. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, 2006.

[35] Mark Culp. spa: Semi-supervised semi-parametric graph-based estimation in R. *Journal of Statistical Software*, 40(10):1–29, 2011.

[36] Leon Danon, Albert Díaz-Guilera, Jordi Duch, and Alex Arenas. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(9):P09008, 2005.

[37] P. De, A. Singh, T. Wong, W. Yacoub, and A. Jolly. Sexual network analysis of a gonorrhoea outbreak. *Sexually Transmitted Infections*, 80:280–285, 2004.

[38] Mosef Draief and Laurent Massouilé. *Epidemics and Rumours in Complex Networks*. Number 369 in London Mathematical Society Lecture Notes Series. Cambridge University Press, 2009.

[39] Niklas Elmqvist and Jean-Daniel Fekete. Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines. *IEEE Transactions on Visualization and Computer Graphics*, 16(3):439–454, 2010.

[40] Paul Erdős and Alfréd Rényi. On random graphs I. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959.

[41] Paul Erdős and Alfréd Rényi. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Science*, 5:17–61, 1960.

[42] Stephen E. Fienberg and Stanley Wasserman. Discussion of "An exponential family of probability distributions for directed graphs" by Holland and Leinhardt. *Journal of the American Statistical Association*, 76(373):54–57, 1981.

[43] Santo Fortunato and Marc Barthélemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41, 2007.

[44] J. H. Fowler and N. A. Christakis. The dynamic spread of happiness in a large social network: Longitudinal analysis over 20 years in the Framingham heart study. *British Medical Journal*, 337:a2338, 2008.

[45] Ove Frank and David Strauss. Markov graphs. *Journal of the American Statistical Association*, 81(395):832–842, 1986.

[46] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[47] T. M. J. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. *Software - Practice and Experience*, 21(11):1129–1164, 1991.

[48] C. Geyer and E. Thompson. Constrained monte carlo maximum likelihood for dependent data. *Journal of the Royal Statistical Society Series B*, 54:657–699, 1992.

[49] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[50] Anna Goldenberg, Alice X. Zheng, Stephen E. Fienberg, and Edoardo M. Airoldi. A survey of statistical network models. *Foundations and Trends in Machine Learning*, 2:129–233, 2010.

[51] Isobel Claire Gormley and Thomas Brendan Murphy. A mixture of experts latent position cluster model for social network data. *Statistical Methodology*, 7(3):385 – 405, 2010.

[52] M. S. Handcock, A. E. Raftery, and J. M. Tantrum. Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series A*, 170(2):1–22, 2007.

[53] Mark S. Handcock, David R. Hunter, Carter T. Butts, Steven M. Goodreau, Pavel N. Krivitsky, and Martina Morris. *statnet: Software Tools for the Statistical Modeling of Network Data*. Seattle, WA, 2003. version 2.6 . Project home page at `http://statnet.org`.

[54] Mark S. Handcock, David R. Hunter, Carter T. Butts, Steven M. Goodreau, Pavel N. Krivitsky, and Martina Morris. *ergm: A Package to Fit, Simulate and Diagnose Exponential-Family Models for Networks*. Seattle, WA, 2011. version 2.4-2.

[55] Mark S. Handcock, Garry Robins, Tom A. B. Snijders, Jim Moody, and Julian Besag. Assessing degeneracy in statistical models of social networks. *Journal of the American Statistical Association*, 76:33–50, 2003.

[56] Steve Hanneke, Wenjie Fu, and Eric P. Xing. Discrete temporal models of social networks. *Electronic Journal of Statistics*, 4:585–605, 2010.

[57] I. Herman, G. Melancon, and M. S. Marshall. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):24–43, 2000.

[58] P. Hoff, A.E Raftery, and M. S. Handcock. Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97(460):1090–1098, 2002.

[59] Peter Hoff. Bilinear mixed-effects models for dyadic data. *Journal of the American Statistical Association*, 100(469):286–295, 2005.

[60] Peter Hoff. Modeling homophily and stochastic equivalence in symmetric relational data. In J. C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 657–664. MIT Press, Cambridge, MA, 2008.

[61] P. W. Holland, K. B. Laskey, and S. Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109 – 137, 1983.

[62] Paul W. Holland and Samuel Leinhardt. An exponential family of probability distributions for directed graphs. *Journal of the American Statistical Association*, 76(373):pp. 33–50, 1981.

[63] Mark Huisman and Marijtje A. J. van Duijn. Software for social network analysis. In Peter J. Carrington, John Scott, and Stanley Wasserman, editors, *Models and methods in social network analysis*. Cambridge University Press, 2004.

[64] David R. Hunter, Steven M. Goodreau, and Mark S. Handcock. Goodness of fit of social network models. *Journal of the American Statistical Association*, 103(481):248–258, 2008.

[65] David R. Hunter and Mark S. Handcock. Inference in curved exponential family models for networks. *Journal of Computational and Graphical Statistics*, 15:565–583, 2006.

[66] Michael I. Jordan. *Learning in graphical models*. Adaptive computation and machine learning. MIT Press, 1998.

[67] Michael I. Jordan. Graphical models. *Statistical Science*, 19(1):pp. 140–155, 2004.

[68] T. Kamada and S. Kawai. An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31:7–15, 1989.

[69] Michael Kaufmann and Dorothea Wagner. *Drawing Graphs: Methods and Models*, volume 2025 of *Lecture Notes in Computer Science*. Springer, 2001.

[70] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46:668–677, 1999.

[71] Valdis E. Krebs. Mapping networks of terrorist cells. *Connections*, 24:43–52, 2002.

[72] Pavel N. Krivitsky. Exponential-family random graph models for valued networks. Technical report, Pennsylvania State University, 2011. Available on arXiv:1101.1359v1.

[73] Pavel N. Krivitsky and Mark S. Handcock. Fitting latent cluster models for networks with latentnet. *Journal of Statistical Software*, 24(5):1–23, 2008.

[74] Pavel N. Krivitsky and Mark S. Handcock. *latentnet: Latent position and cluster models for statistical networks*, 2010.

[75] Pavel N. Krivitsky, Mark S. Handcock, and Martina Morris. Adjusting for network size and composition effects in exponential-family random graph models. *Statistical Methodology*, 8(4):319 – 339, 2011.

[76] Pavel N. Krivitsky, Mark S. Handcock, Adrian E. Raftery, and Peter D. Hoff. Representing degree distributions, clustering, and homophily in social networks with latent cluster random effects models. *Social Networks*, 31(3):204–213, 2009.

[77] Pierre Latouche, Etienne Birmelé, and Christophe Ambroise. Overlapping stochastic block models with application to the french political blogosphere. *Annals of Applied Statistics*, 5(1):309–336, 2011.

[78] David Lazer, Ines Mergel, and Allan Friedman. Co-citation of prominent social network articles in sociology journals: The evolving canon. *Connections*, 29:43–64, 2009.

[79] F. Lorrain and H. C. White. Structural equivalence of individuals in social networks. *Journal of the Mathematical Sociology*, 1(1):49–80, 1971.

[80] Russell Lyons. The spread of evidence-poor medicine via flawed social-network analysis. *Statistics, Politics, and Policy*, 2(1):Article 2, 2011.

[81] Amin Mantrach, Nicolas van Zeebroeck, Pascal Francq, Masashi Shimbo, Hugues Bersini, and Marco Saerens. Semi-supervised classification and betweenness computation on large, sparse, directed graphs. *Pattern Recognition*, 44(6):1212 – 1224, 2011.

[82] Aaron McDaid and Neil J. Hurley. Detecting highly overlapping communities with model-based overlapping seed expansion. In Nasrullah Memon and Reda Alhajj, editors, *International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 112–119. IEEE Computer Society, 2010.

[83] Nicolai Meinshausen and Peter Bühlmann. High-dimensional graphs and variable selection with the Lasso. *Annals of Statistics*, 34(3):1436–1462, 2006.

[84] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006.

[85] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pages 849–856. MIT Press, 2001.

[86] Krzysztof Nowicki and Tom A. B. Snijders. Estimation and prediction of stochastic blockstructures. *Journal of the American Statistical Association*, 96(455):1077– 1087, 2001.

[87] G. Palla, I. Derényi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

[88] Ian Porteous, Arthur Asuncion, David Newman, Padhraic Smyth, Alexander Ihler, and Max Welling. Fast collapsed Gibbs sampling for latent Dirichlet allocation. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 569–577, New York, NY, USA, 2008. ACM.

[89] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2010. ISBN 3-900051-07-0.

[90] Adrian E. Raftery, Xiaoyue Niu, Peter D. Hoff, and Ka Yee Yeung. Fast inference for the latent space network model using a case-control approximate likelihood. *Journal of Computational and Graphical Statistics*, To appear, 2012.

[91] E. M. Reingold and J. S. Tilford. Tidier drawings of trees. *IEEE Transactions on Software Engineering*, 7:223–228, 1981.

[92] Garry Robins, Pip Pattison, Yuval Kalish, and Dean Lusher. An introduction to exponential random graph (p^*) models for social networks. *Social Networks*, 29(2):173 – 191, 2007.

[93] Garry Robins, Tom A. B. Snijders, Peng Wang, and Mark S. Handcock. Recent developments in exponential random graph (p^*) models for social networks. *Social Networks*, 29(2):192–215, 2006.

[94] Karl Rohe, Sourav Chatterjee, and Bin Yu. Spectral clustering and the high-dimensional stochastic blockmodel. *Annals of Statistics*, 39(4):1878–1915, 2011.

[95] J. N. Rosenquist, J. Murabito, J. H. Fowler, and N. A. Christakis. The spread of alcohol consumption behavior in a large social network. *Annals of Internal Medicine*, 152(7):426–433, 2010.

[96] Michael Salter-Townshend. *VBLPCM: Variational Bayes Latent Position Cluster Model for networks.*, 2012. R package version 2.0.

[97] Michael Salter-Townshend and Thomas Brendan Murphy. Variational Bayesian inference for the latent position cluster model. In *NIPS Workshop on Analyzing Networks and Learning with Graphs*, 2010.

[98] P. Sargolzaei and F. Soleymani. Pagerank problem, survey and future research directions. *International Mathematical Forum*, 5(19):937–956, 2010.

[99] Ted E. Senator, Henry G. Goldberg, Jerry Wooton, Matthew A. Cottini, A. F. Umar Khan, Christina D. Klinger, Winston M. Llamas, Michael P. Marrone, and Raphael W. H. Wong. The FinCEN artificial intelligence system: Identifying potential money laundering from reports of large cash transactions. In Jan Atkins and Howard Shrobe, editors, *Proceedings Of The Seventh Conference On Innovative Applications Of Artificial Intelligence*, pages 156–170, 1995.

[100] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker. Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome Research*, 13:2498–2504, 2003.

[101] Jianbo Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888 –905, 2000.

[102] Susan Shortreed, Mark S. Handcock, and Peter Hoff. Positional estimation within a latent space model for networks. *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences*, 2(1):24 – 33, 2006.

[103] Paolo Simonetto, David Auber, and Daniel Archambault. Fully automatic visualisation of overlapping sets. *Computer Graphics Forum*, 28(3):967–974, 2009.

[104] Tom A. B. Snijders. Markov chain Monte Carlo estimation of exponential random graph models. *Journal of Social Structure*, 3(2):1–40, 2002.

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

[106] K. A. Stephenson and M. Zelen. Rethinking centrality: Methods and examples. *Social Networks*, 11:1–37, 1989.

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

[108] Amanda L. Traud, Eric D. Kelsic, Peter J. Mucha, and Mason A. Porter. Comparing community structure to characteristics in online collegiate social networks. *SIAM Review*, 53(3):526–543, 2011.

[109] M. A. J. van Duijn, K. J. Gile, and M. S. Handcock. A framework for the comparison of maximum pseudo-likelihood and maximum likelihood estimation of exponential family random graph models. *Social Networks*, 31:52–62, 2009.

[110] Marijtje A. J. van Duijn, Tom A. B. Snijders, and Bonne J. H. Zijlstra. p_2: a random effects model with covariates for directed graphs. *Statistica Neerlandica*, 58(2):234–254, 2004.

[111] Various. *RSiena: Siena - Simulation Investigation for Empirical Network Analysis*, 2011. R package version 1.0.12.167.

[112] Tatiana von Landesberger, Arjan Kuijper, Tobias Schreck, Jrn Kohlhammer, Jarke J van Wijk, Jean-Daniel Fekete, and Dieter W Fellner. Visual Analysis of Large Graphs: State-of-the-Art and Future Research Challenges. *Computer Graphics Forum*, 30(6):1719–1749, 2011.

[113] Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17:395–416, 2007.

[114] M. J. Wainwright and M. I. Jordan. *Graphical Models, Exponential Families, and Variational Inference*. Now Publishers, 2008.

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

[116] S. Wasserman and K. Faust. *Social network analysis: Methods and applications*. Cambridge Univ Press, 1994.

[117] Stanley Wasserman and Philippa Pattison. Logit models and logistic regressions for social networks: I. An introduction to Markov graphs and p^*. *Psychometrika*, 61:401–425, 1996. 10.1007/BF02294547.

[118] Stanley Wasserman, Garry Robins, and Douglas Steinley. Statistical models for networks: a brief review of some recent research. In *Proceedings of the 2006 conference on Statistical network analysis*, ICML'06, pages 45–56, Berlin/Heidelberg, 2007. Springer-Verlag.

[119] Scott White and Padhraic Smyth. A spectral clustering approach to finding communities in graphs. In *Proceedings of the Fifth SIAM International Conference on Data Mining*, volume 119, pages 274–285, 2005.

[120] Jianmin Wu, Tea Vallenius, Kristian Ovaska, Jukka Westermarck, Tomi P Makela, and Sampsa Hautaniemi. Integrated network analysis platform for protein-protein interactions. *Nature Methods*, 6:75–77, 2009.

[121] Eric P. Xing, Wenjie Fu, and Le Song. A state-space mixed membership blockmodel for dynamic network tomography. *Annals of Applied Statistics*, 4(2):535–566, 2010.

[122] W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4):452–473, 1977.

[123] Dengyong Zhou and Bernhard Schölkopf. A regularization framework for learning from graph data. In *ICML Workshop on Statistical Relational Learning and Its Connections to Other Fields*, page 132137, 2004.

[124] Dengyong Zhou, Bernhard Schölkopf, and Thomas Hofmann. Semi-supervised learning on directed graphs. *Biological Cybernetics*, 17:1633–1640, 2005.

[125] Xiaojin Zhu. Semi-supervised learning literature survey (revised edition). Technical report, Technical Report 1530, Department of Computer Sciences, University of Wisconsin, Madison, 2008.

[126] Xiaojin Zhu and Andrew B. Goldberg. Introduction to semi-supervised learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 3(1):1–130, 2009.