Complex Networks: Structure and Inference

by

Alec Kirkley

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Physics)
in the University of Michigan
2021

Doctoral Committee:

Professor Mark Newman, Chair
Associate Professor Robert Deegan
Associate Professor Danai Koutra
Professor David Lubensky
Associate Professor Xiaoming Mao

Alec Kirkley
akirkley@umich.edu
ORCID iD: 0000-0001-9966-0807

Dedication

To Shihui, the love of my life.

Acknowledgments

I am incredibly fortunate to have had such a supportive group of people behind me during my graduate studies. My thesis advisor, Mark Newman, has provided invaluable guidance on every aspect of the academic experience, from the execution of technical fundamentals to the conceptualization and organization of research ideas to the presentation of information to students and colleagues. His openness, patience, and generosity have made working with him an extremely fun, engaging, and rewarding experience. He is a role model for me in my academic pursuits, and in my future endeavors I will strive to uphold his standards of elegance, rigor, and pragmatism that make him a truly unique scholar.

I also owe a debt of gratitude to my other collaborators. George Cantwell and Jean-Gabriel Young comprise the rest of the brilliant Newman research group that I've been lucky enough to work with and learn from, and I thank them both for their mentorship and friendship. Gourab Ghoshal, who introduced me to network science as I commenced my academic journey, has remained a close collaborator during my graduate studies, and I thank him for all of his help and guidance. I also thank Greg van Anders and Danai Koutra, whom I have had the pleasure of working with on other scientific pursuits during my time at UM, and who have provided support for me during my grant- and job-seeking efforts. And on that note, I thank the National Defense Science and Engineering Graduate Fellowship (NDSEG) program for fund-

ing the final three years of my graduate studies.

Finally I thank my amazing family and friends, from Ann Arbor to Rochester to Changchun, in particular my loving mom, dad, and wife Shihui, for their unconditional support and patience with me during this process.

Table of Contents

List of Figures

List of Tables

List of Appendices

Abstract

From the spread of disease across a population to the dispersion of vehicular traffic in cities, many real-world processes are driven by lots of small components that interact in simple ways at small scales to produce nontrivial large-scale effects. Probing the fundamental mechanisms that govern such systems— broadly called "complex systems"—is crucial for control, design, and intervention relevant to these processes. Networks, mathematical objects composed of nodes attached in pairs by edges, provide a very useful representation of such systems, and thus modelling networks is of critical importance for understanding real-world complex systems. In this thesis, I examine two different aspects of network modelling: (1) characterizing structure in networks with metadata, and (2) developing scalable, accurate, and interpretable inference techniques for real-world network data.

I approach the problem of characterizing structure in networks with metadata from two different perspectives. First, I discuss new measures for characterizing the structure of signed networks with positive and negative edge signs representing amity and enmity respectively. Signed networks are hypothesized to display structural regularity (balance) as a result of certain configurations of edge signs being more common than others—for instance, the friend of my enemy should be my enemy. I show that we can develop intuitive measures of balance in signed networks that capture

long-range correlations, demonstrating that real networks are indeed significantly balanced using these measures, and that these measures can be used to impute missing data. Second, I move on to explore how we can measure diversity at multiple scales in networks with node metadata that take the form of distributions. I detail a general information theoretic framework for this task, illustrating new insights it can give us through example applications involving demographic data across spatially contiguous regions.

With regards to inference, I first describe a new message passing algorithm for fast approximate inference in probabilistic graphical models on networks with short loops. I derive a self-consistent set of message passing equations using a decomposition of the network into generalized neighborhoods surrounding each node, and extend these equations to compute thermodynamic quantities of interest in spin systems including the specific heat and entropy. I then outline an information theoretic clustering algorithm to summarize posterior distributions over community labellings by identifying representative partitions. I cluster sampled community partitions around representative "modal" partitions using an efficient algorithm based on the minimum description length principle, finding that a variety of distinct modal partitions with different interpretations exist for example networks.

Altogether, these contributions allow for new insights about real-world network data that are obscured by existing measures and methods, and provide a toolkit for researchers to explore the properties of a wide variety of complex systems in an efficient and principled manner. The work in this thesis is intentionally motivated by very fundamental principles, and so provides a starting point for a variety of future domain- and application-specific optimizations.

Chapter 1.

Introduction

When we analyze a real-world system—physical, biological, social, technological, or otherwise—we commonly find that pairwise relationships between distinct fundamental units in the system drive its structure and dynamics. Magnetic phenomena in materials are governed by correlations between the dipole moments of pairs of atomic spins. Cellular processes in an organism are mediated by interactions between pairs of proteins. Ideas and diseases spread through a population via contacts between pairs of people. Electric power is transmitted across countries between pairs of power stations. And goods are exchanged between pairs of nations in international trade. The list goes on and on. A network, which in its most basic form is a collection of individual units, or *nodes*, that are linked together with pairwise connections, or *edges*, captures precisely this notion of pairwise dependence between entities, and networks are thus a useful representation of a variety of real-world systems. As a result, there has been a huge effort to expand the set of theoretical tools for analyzing networks over the last few decades [1–14], and network science has influenced an array of fields as diverse as neuroscience, finance, economics, archaeology, education,

political science, and medicine [15–27].

Despite the numerous theoretical advances in network modelling over recent years, there are many practical applications for which existing network analysis tools are not sufficient. In many cases, we want to use as much information as is available to us in our analysis of a system, and the data we can obtain generally includes much more about the system than just the structural topology that we can represent with nodes and edges. In most situations, we can augment these nodes and edges with metadata, which we can couple to the topology of the system's underlying network to refine our understanding of its function. In the first half of this thesis, I will focus on methods and measures I have developed that allow us to incorporate metadata into our analysis of networks, bringing new insights to a variety of phenomena.

In addition to metadata, another property of real systems that precludes the application of existing network analysis techniques is the sheer scale of the data they produce. In most early applications of network-style analysis [28–30], the datasets were collected by hand and rarely exceeded a few hundred nodes. For such data, many standard algorithmic techniques such as matrix inversion or complete enumeration of shortest paths are feasible. Nowadays, with the World Wide Web and other digital technologies contributing massive, rich datasets with networks of sizes in the hundreds of thousands to millions, and even billions, to be studied [31–33], these tried and true methods are rendered useless for many practical applications of interest. Moreover, due to the inherently high dimensionality of network data, even for small networks it is challenging to intuitively grasp the complex output of many network analysis tools (for example, Bayesian posterior distributions over community structures, as we will discuss in Chapter 5). In the second half of this thesis, I will discuss two methods I have developed to enable scalable, interpretable inference with real

network data, helping to close the gap between the theory and application of network analysis.

But before we delve into more detailed results, I will give a brief overview of some basic definitions, measures, models, and inference methods involved in the theoretical analysis of networks. The purpose of this chapter is to facilitate a smooth transition into more sophisticated methods by reviewing relevant foundational concepts in network science. This being said, it is not meant to be a comprehensive introduction to network analysis, for which there are thorough, dedicated texts [1, 2]. First, we will go over some definitions.

1.1. Network definitions and representations

As mentioned, the most elementary form of a network consists of a set of n nodes, which we denote with V, connected in pairs by m edges, the set of which we denote E. These node and edge sets can be combined into a single mathematical object, a *graph*, denoted by $G = (V, E)$. Rather than focusing on the entire graph G, it is sometimes easier to only consider a subset of the nodes V and edges E, which we refer to as a *subgraph* of G. Associated with the graph G is an $n \times n$ *adjacency matrix* $\mathbf{A}$ that encodes the positions of the edges E in terms of the nodes in V that they connect. More concretely, the entry A_{ij} of the adjacency matrix is given by

$$A_{ij} = \begin{cases} 1, \text{ node } i \in V \text{ is connected to node } j \in V \text{ by edge } (i,j) \in E, \\ 0, \text{ otherwise.} \end{cases} \tag{1.1}$$

In this case, we do not endow the edges in E with any direction—the tie (i, j) is mutual and equivalent to (j, i)—or weight—the graph and its corresponding adjacency matrix simply denote the existence of edge (i, j)—and so the network is considered *undirected* and *unweighted*. An example of an undirected, unweighted graph is shown in Fig. 1.1A.

We can augment a graph G with additional information in any number of ways, the most common being directions, weights, and/or types for the edges. For directed graphs, the edge (i, j) and non-zero A_{ij} denote the existence of an edge pointing from node i to node j. (Some people prefer the reverse notation, that is to let A_{ij} indicate the presence or absence of an edge from j to i.) For weighted edges, the adjacency matrix $\mathbf{A}$ is no longer binary, and A_{ij} encodes the weight of the edge between nodes i and j. (In some case it is convenient to use a separate matrix to denote the edge weights, in addition to a binary adjacency matrix to encode the existence of an edge.) We can adjust our notation for the edge set E to accommodate these additional features by extending each tuple (i, j) to include the attributes for the edge (i, j). Examples of a weighted, directed network and an undirected network with multiple edges types are shown in Fig. 1.1B and Fig. 1.1C respectively.

In this thesis we will also make mention of *bipartite* networks, which consist of two sets of nodes, call them P and R, such that edges only run between sets P and R—that is, any edge has one incident node in set P and one in set R. A *projection* of this bipartite graph onto set R then consists only of nodes in set R, and two nodes are connected if and only if they shared at least one common neighbor in set P in the original, un-projected network (and vice-versa for projection onto set P). Such a representation is commonly used, for instance, in modelling collaboration patterns among researchers. In this case, set R might be the researchers, set P the research

papers, and an edge runs between researcher $r \in R$ and paper $p \in P$ if researcher r was on paper p. The projection of this network onto the researchers is then the "one-mode" network of researchers R such that an edge connects researchers $r \in R$ and $s \in R$ if and only if researcher r and researcher s collaborated on at least one paper $p \in P$. Likewise, the projection of this network onto the papers P is the one-mode network of papers P such that an edge connects papers $p \in P$ and $q \in P$ if and only if there was at least one researcher $r \in R$ who was on both paper p and paper q.

Other common extensions of basic graphs that are used in network modelling include graphs with scalar or vector attributes for the nodes, graphs with self-edges (edges connecting nodes with themselves), multilayer networks with multiple edge types and possible interaction between network layers [34–36], hypergraphs in which a single edge may connect more than two nodes [37], and networks of simplicial complexes that can represent nested sets of ties with multiple dimensionalities [38].

In terms of nomenclature, one generally uses the term "network" to refer to a system of interest that can be represented mathematically with a graph, although in many contexts the terms "network" and "graph" are used interchangeably. However, network science and graph theory are quite distinct subject areas in terms of their research focuses [39]. Graph theory involves the rigorous proof of mathematical properties of graphs, which commonly requires the graph to have symmetries and other idealized features that are not present in real-world data. On the other hand, network science primarily focuses on quantifying and modelling the structure and dynamics seen in real-world network data, which often does not have nice analytical properties. That being said, network science does borrow many ideas from graph theory, particularly for its formal analytical results [1]. The research I present in this thesis is much better classified under network science than graph theory.

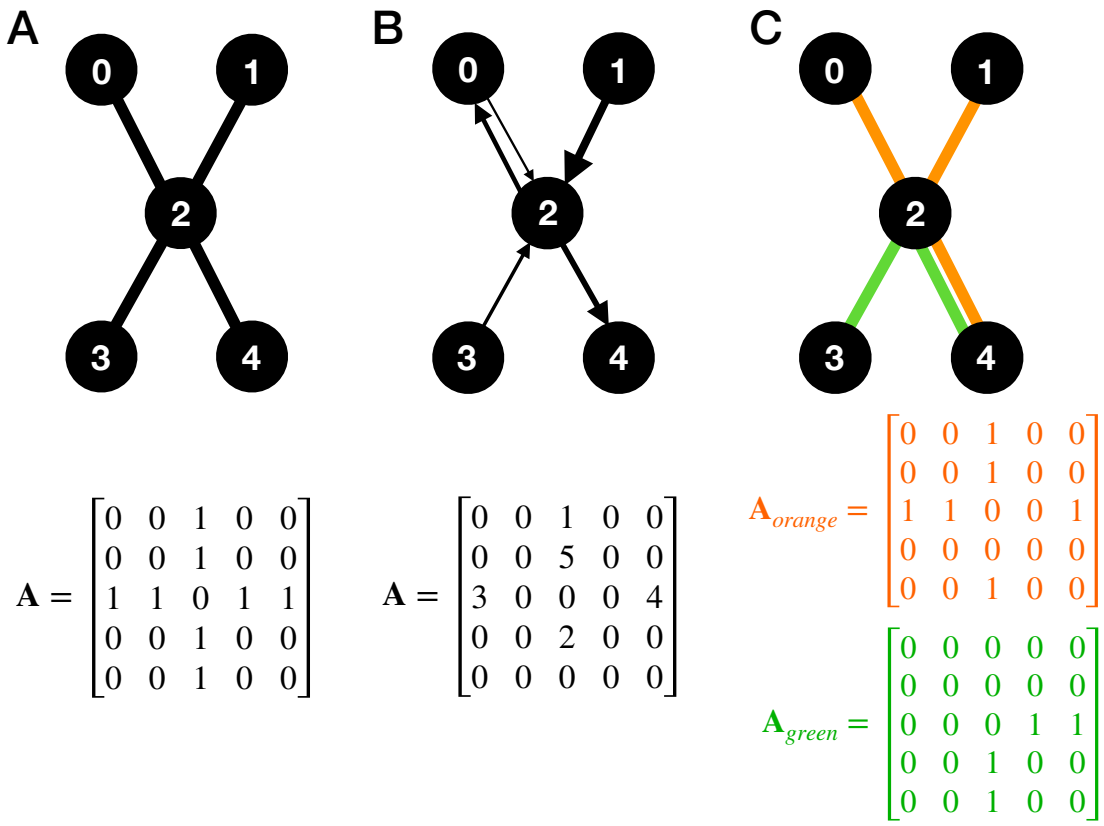

Fig. 1.1. Examples of graphs and their corresponding adjacency matrices. **(A)** An undirected, unweighted graph on five nodes $V = \{0, 1, 2, 3, 4\}$ with four edges $E = \{(0, 2), (1, 2), (2, 3), (2, 4)\}$. **(B)** A directed, weighted graph on the same five nodes, with five (weighted) edges $E = \{(0, 2, 1), (2, 0, 3), (1, 2, 5), (3, 2, 2), (2, 4, 4)\}$. **(C)** An undirected network on the same five nodes with five edges $E = \{(0, 2, \text{orange}), (1, 2, \text{orange}), (2, 3, \text{green}), (2, 4, \text{green}), (2, 4, \text{orange})\}$, along with the adjacency matrices associated with the green and orange edges.

1.2. Structural properties of real-world networks

Armed with the definitions above, we can now define measures that capture different aspects of the structure of networks, such as how central a node is or how naturally the network breaks up into subgroups of tightly connected nodes. Empirical measurements of these quantities can aid in summarizing the complex structure of a network, as well as provide insight into the principles underlying its organization and function. The observed values of many of these properties differ substantially in real-world networks from what one would expect in simple idealized models of graphs, which has inspired a huge research effort to develop models that accurately capture key structural characteristics of real networks.

1.2.1. Degrees

Perhaps the simplest statistic one may be interested in when analyzing a network is the *degree* of a node, or how many connections it has to other nodes. The degree is a very intuitive proxy for the importance of a node, as nodes that are more well connected may play a more active or central role in the function of the network. The degree k_i of a node i in an undirected, unweighted network is given by

$$k_i = \sum_{j=1}^{n} A_{ij} = |\partial_i|, \tag{1.2}$$

where $\partial_i \subseteq V$ is the set of nodes that are connected to i by an edge, also known as its *neighborhood*. For notational convenience, we will assume henceforth that summations over node indices are over the entire set V of n nodes, and so we will omit the limits. When the network is undirected and unweighted, k_i is the total number of

edges incident to node i.

There are many common alternative measures to degree that encode slightly different information and can be applied to weighted and directed networks. For example, for directed networks it makes sense to assign two separate degree values to each node i, an *in-degree*, $k_i^{in} = \sum_j A_{ji}$, and an *out-degree*, $k_i^{out} = \sum_j A_{ij}$, which count respectively the number of edges pointing towards node i and the number of edges pointing outwards from node i. For weighted networks, one can assign a *strength* value to each node i, equal to the sum of the weights of the edges incident on node i (which can be extended to in- and out-strengths for directed, weighted graphs in a manner analogous to the degrees).

We can gain insight into the global structure of a network by looking at its empirical *degree distribution* P_k, the fraction of nodes that have degree k. Since each node can only have one degree value (we are only considering undirected networks here), P_k is a properly normalized probability mass function over the non-negative integers. From this we can compute the average degree,

$$\langle k \rangle = \sum_{k=0}^{\infty} k P_k = \frac{2m}{n}, \tag{1.3}$$

and the degree variance,

$$\langle k^2 \rangle - \langle k \rangle^2 = \sum_{k=0}^{\infty} k^2 P_k - \langle k \rangle^2, \tag{1.4}$$

which are both useful quantities for understanding the large-scale organization of the network. If the average degree $\langle k \rangle$ is large, then we know that the network has a high density of edges, and so the typical node is highly connected. On the other hand, if the

degree variance is large, we know that there is a high level of variation in connectivity among nodes: some nodes are much more well connected than others.

There are two characteristic properties associated with the degree distributions of many real-world networks which are critical for a wide range of observed functionalities in these systems. The first is *sparsity*, or that a vanishing fraction of possible connections are present in the network as it grows. Technically, there are $\binom{n}{2} \sim \mathrm{O}(n^2)$ possible connections in an undirected graph (where $\sim \mathrm{O}(\cdot)$ indicates Big O notation [40]), and so we only require that $m \sim \mathrm{o}(n^2)$ grows at less than this rate to have sparsity. However, in most cases, when we say that a network is "sparse", we are indicating that $m \sim \mathrm{O}(n)$, or that the average degree $\langle k \rangle \sim \mathrm{O}(1)$ is roughly constant. Of course, one cannot rigorously determine based on a single observation of a network at one particular instance in time how exactly the number of edges scales with the number of nodes, and so "sparsity" is not really a rigorous concept for networks in many cases. But roughly speaking, if the average degree $\langle k \rangle$ is much less than the size of the network n, we say in practice that the network is sparse. The sparsity of a network has very important effects on its functionality, since the majority of node pairs are not connected by an edge, and more complex paths need to be utilized in order for anything to traverse through the network. (Surprisingly, however, many real networks display a 'small-world' property, where on average we can hop from one node to another in a number of steps that is much much smaller than the size of the system [29, 41, 42].) This sparsity also becomes critical for modelling and analyzing real networks, as it allows us to make various analytical approximations [1] and generally improves performance of graph algorithms.

The second characteristic observed empirically in the degree distributions of many real-world networks is that they have *heavy tails*, or that their tails are not exponentially

bounded (unlike Binomial or Poisson distributions, which describe the degree distributions of networks generated by purely random placement of edges; more on this to follow). Practically, this means that although most nodes may have quite low degrees, there are a non-negligible number of nodes that have an extremely high number of connections. These "hubs" play a critical role in the global connectivity of the network, allowing networks with heavy-tailed degree distributions to be more resilient to random node failures [43], have short path lengths for navigation between nodes [44], and respond well to simple immunization strategies [45, 46], among other advantages [9].

In many cases, these observed heavy-tailed degree distributions (approximately) follow the power-law form

$$P_k \sim k^{-\gamma}, \tag{1.5}$$

for some exponent γ, a distribution which permits simple analytical treatment and has interesting theoretical properties. In fact, many networks display power-law degree distributions with $\gamma \in (2, 3)$, in which case the corresponding degree variance (Eq. 1.4) is infinite! (Of course, this divergence cannot be observed in the degree distribution of a real network, but it has interesting implications for theoretical models of networks. In practice, one normally assumes that there is an exponential cutoff to the degree distribution that permits the empirically observed finite moments.) In part, empirically observed power-law degree distributions were what attracted a huge wave of physicists to study networks in the late 2000's [47, 48], although this property had garnered interest from scientists decades earlier in bibliometric studies [49, 50]. It is currently a highly debated topic whether or not real degree distributions rigorously

follow the power-law form [51–53], but for practical purposes, it is the heavy-tailed nature of the degree distribution that has an important impact on the structure and function of real networks. Examples of networks with power law degree distributions of varying exponents γ are shown in Fig. 1.2A. For $\gamma = 3$ (moderate degree variance), we can see that there are a few nodes of moderately high degree, while for $\gamma = 2$ (high degree variance) there are a number of nodes with moderately high degrees and a few with extremely high degrees.

1.2.2. Walks

Going beyond the immediate neighborhood of a node, we can compute quantities related to *walks* in the network, which consist of traversals of the nodes along the edges connecting them. For example, a walk of length 3 in the network shown in Fig. 1.1A might consist of starting at node 2, moving to node 1 along edge $(1, 2)$, moving back to node 2 along this same edge, and then moving to node 4 along the edge $(2, 4)$. Note that in a walk we can traverse through a given node or edge multiple times, as in this example. This is in contrast to a *path* in a graph, in which all nodes (and consequently edges) are distinct. An undirected network is called *connected* if there is at least one path between all pairs of nodes in the network. In a network there may be lots of ways to walk from a source node i to a target node j in l steps. In this example, for instance, we could have instead gone across edge $(0, 2)$ to node 0 in the first step, then back to node 2 and finally to node 4, a walk which would have also consisted of $l = 3$ steps with start node 2 and end node 4. In any such walk, we are traversing an edge at each step, and so require that for the sequence of $l + 1$ visited nodes in the walk, $\{i, p_1, p_2, ..., p_{l-1}, j\}$, each consecutive pair of nodes (p_t, p_{t+1}) has a common

edge, or equivalently that $A_{p_t,p_{t+1}} = 1$. Therefore, such a sequence of nodes comprises a possible walk from i to j if and only if $A_{ip_1}A_{p_1p_2}\cdots A_{p_{l-2}p_{l-1}}A_{p_{l-1}j} = 1$. Summing over all possible sets of intermediate nodes gives us the number of possible walks along the network from i to j of length l, thus

$$\begin{aligned}\text{\# of walks of length } l \text{ between } i \text{ and } j &= \sum_{p_1,p_2,\ldots,p_{l-1}} A_{ip_1}A_{p_1p_2}\cdots A_{p_{l-2}p_{l-1}}A_{p_{l-1}j} \\ &= (\mathbf{A}^l)_{ij}. \end{aligned} \tag{1.6}$$

Using the same concept, we can count the number of *closed walks* (walks that start and end at the same node) of a given length l with powers of the adjacency matrix. Letting $i = j$ in Eq. 1.6 and summing over all nodes i, we get

$$\text{\# of closed walks of length } l = \text{Tr}(\mathbf{A}^l). \tag{1.7}$$

Note that this expression counts connected subgraphs of G multiple times, as it accounts for all possible ways in which we could traverse a given set of adjacent edges. For instance, if we want to count the number of triangles in the network, we need to use a modified expression, $\text{Tr}(\mathbf{A}^3)/6$, which divides out the $3! = 6$ redundant traversal orders of each triangle that are counted in Eq. 1.7. We distinguish closed walks from a similar concept, *simple cycles*, which are the subset of closed walks that do not visit any node twice, other than the start/end node, which is visited exactly twice. Counting simple cycles is perhaps more informative about network structure as they do not contain redundant edges. However, simple cycles are much more difficult to count than closed walks (the exception being triangles, as we have shown), and so typically when constructing network measures one uses counts of closed walks instead

(see Ch. 2 for an example).

For undirected networks, it is useful to consider the diagonalization of the symmetric matrix $\mathbf{A}$

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}, \tag{1.8}$$

where $\mathbf{U}$ is an orthogonal matrix whose k-th column is the k-th eigenvector of $\mathbf{A}$, and $\mathbf{\Lambda}$ is a diagonal matrix of the (real) eigenvalues $\{\lambda_k\}_{k=1}^n$ of $\mathbf{A}$. We can then write Eq. 1.7 as

$$\text{\# of closed walks of length } l = \text{Tr}(\mathbf{U}\mathbf{\Lambda}^l\mathbf{U}^{-1}) = \text{Tr}(\mathbf{\Lambda}^l) = \sum_k \lambda_k^l. \tag{1.9}$$

Eq. 1.9 demonstrates that we can recover important information about the topology of a network from the eigenvalues of its adjacency matrix. In fact, there is an entire branch of graph theory, *spectral graph theory*, devoted to understanding the spectra of the matrices associated with graphs [54].

1.2.3. Clustering

Of the types of simple cycles present in networks, triangles are perhaps the most important, as they point to the tendency of links to form in a transitive manner—if i and j are connected, and j and k are connected, is it likely that i and k are also connected? The most straightforward measure of this tendency is aptly called the *transitivity* of the network, and is given by

$$T = \frac{\text{\# of paths of length 2 that are part of a triangle}}{\text{\# of paths of length 2}} = \frac{\text{Tr}(\mathbf{A}^3)}{\sum_i k_i(k_i-1)}. \tag{1.10}$$

Eq. 1.10 quantifies precisely the intuitive notion of transitivity we have mentioned: of all the paths (non-self overlapping walks) of length 2, it tells us in what fraction the start and endpoints are also connected. The transitivity is one measure on which we base the claim that a network is "clustered". Examples of networks with varying levels of transitivity are shown in Fig. 1.2B. We can see that the network with low clustering ($T = 0.03$) appears more or less like a random ball of edges, while the network with higher clustering ($T = 0.24$) has a visibly high density of triangles.

As one may expect, real social networks display a high level of transitivity [55, 56], as it is likely that if persons i and j are acquainted, and persons j and k are acquainted, that persons i and k will also be acquainted. The extent to which social networks in the real-world are clustered is actually quite remarkable: it is not uncommon for these networks to have values of T around 0.5 or higher [1]. For comparison, consider the situation where one picks friends at random from the population. If I pick c friends at random, and each of my friends does the same, the probability that two of my friends are themselves friends is $c/(n-1)$. For a sparse network, $c \ll n$, and so the transitivity in this case is expected to be vanishingly small for a large network. This is of course a very rough approximation, for one due to the fact that people can have very different numbers of friends, but one would expect that even a more refined model of this sort would produce a transitivity that is well below the observed values of ~ 0.5 in real networks. This high level of clustering has a profound impact on the function of real networks, in particular their resilience to epidemics [57, 58] which tend to spread more quickly with greater clustering.

An alternative measure for assessing the level of clustering in a network is based on the *local clustering coefficient* [42], which counts what fraction of pairs of a given node i's neighbors are themselves connected. There are $k_i(k_i-1)/2$ possible edges between

the neighbors of i, and so this measure takes the form

$$C_i = \frac{\text{\# of pairs of neighbors of } i \text{ that are connected}}{k_i(k_i-1)/2}. \tag{1.11}$$

To assess the global level of clustering in a similar manner as in Eq. 1.10, we can take the average of these local clustering coefficients, thus

$$C = \frac{1}{n}\sum_i C_i. \tag{1.12}$$

Since Eq. 1.12 weights each node equally when taking the average, it tends to be dominated by contributions from low degree nodes, and so may give significantly different results than Eq. 1.10 on the same network. We thus cannot use these measures interchangeably; rather, we can use them both to get a multifaceted picture of the clustering in a network.

1.2.4. Community structure

The degree quantifies the extent to which individual nodes are highly connected, and the various measures of clustering quantify the extent to which networks are tightly connected at the small scale (the level of triangles). But what about the larger scale connectivity structure of a network? One way of tackling this question is to identify natural divisions of a network into modules, called *communities*, in a process called *community detection*. In the most common case, which we will focus on here, community detection consists of partitioning the node set V into disjoint groups (communities) such that there is a high density of connections within communities and a sparser level of connectivity between communities. This partition of the nodes in the

network can be represented by a community assignment vector $\mathbf{g}$, such that g_i is the group to which node i is assigned. Many standard methods for community detection then typically proceed by identifying the partition $\mathbf{g}$ of the network that maximizes some objective (or "quality") function $Q(\mathbf{g})$, which assesses how good the partition is according to some set of criteria.

One natural way to measure the quality of a partition is the compute the *modularity* of the partition, which quantifies the extent to which the density of edges within the communities of the partition exceeds what we would expect in a comparable network with edges rewired completely at random [59]. The modularity $Q(\mathbf{g})$ of a network partition $\mathbf{g}$ is given by

$$Q(\mathbf{g}) = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta_{g_i, g_j}, \tag{1.13}$$

where $\delta_{x,y}$ is the Kronecker delta symbol which equals 1 when $x = y$ and 0 otherwise. This serves the function of restricting the sum to be only over pairs of nodes i, j that are in the same community. Looking at the summand, we can see two contributions. The left hand term A_{ij} indicates the presence/absence of an edge between i and j, with an increase of 1 to the modularity for each present within-community edge. The right hand term $k_i k_j/(2m)$ indicates the number of edges we would expect to see between nodes i and j if the graph were rewired completely random while fixing the degree of each node, and this quantity is subtracted from the modularity. This enforces the idea that the presence of within-community edges that are highly likely based on pure chance provides little evidence for the quality of a partition. On the other hand, the presence of an unlikely edge within a group will less heavily penalize the modularity, and so the contribution of the summand $A_{ij} - k_i k_j/(2m)$ is greater.

Examples of networks with varying levels of modularity are shown in Fig. 1.2C. The graph with lower modularity ($Q = 0.19$) clearly has some level of division between the two groups, but it is not very clear. On the other hand, the graph with high modularity ($Q = 0.49$) has very well defined group structure, with dense connectvity within the groups and only a few edges between the groups.

We can see that $k_i k_j/(2m)$ is (approximately) the expected number of edges between i and j under random rewiring—or, if $k_i k_j < 2m$ as is usually the case, the probability that there is an edge between i and j–using the following argument. In a model where each node i has the same degree k_i but the edges in the graph are rewired at random, we can think of this node as contributing k_i "stubs", which will be matched up randomly with other nodes' stubs to create the edges. Generating a realization of this model, called the *configuration model* in the network science literature, consists of picking pairs of stubs at random and adding an edge between their corresponding nodes until we exhaust all $2m$ stubs. This specific procedure will create networks that in general have both edges from nodes to themselves as well as multiple edges between nodes, but one can show that the probability of this happening is generally very small [1] and there are more sophisticated algorithms that attempt to overcome this problem [60]. As we rewire the network, there will be k_i chances for node i to connect to node j, each of which will be successful with a probability of approximately $k_j/(2m)$, and so $k_i k_j/(2m)$ gives the expected number of edges between i and j resulting from this rewiring process.

Once we have a quality function $Q(\mathbf{g})$ for a partition $\mathbf{g}$, we proceed by attempting to maximize Q over all possible partitions $\mathbf{g}$. Unfortunately, in most practical applications we cannot exhaustively check Q on all these partitions, as their number increases exponentially with the size n of the network. As an example to illustrate

how difficult exhaustive enumeration of these partitions is, consider a network with $n = 50$ nodes (which is very small compared to the thousands or more we see in typical real networks) that we want to divide into $K = 10$ groups. There are approximately $10^{50}/10! \approx 2.6 \times 10^{43}$ ways we can partition these 50 nodes into 10 groups. The fastest supercomputer in the world (at the time of writing this thesis) can execute around 5×10^{17} operations per second, and so it would take roughly 5×10^{25} seconds for this supercomputer to exhaustively check Q on all partitions $\mathbf{g}$ of this network, which is around 100 million times the age of the universe! In practice, one thus resorts to a variety of optimization techniques to maximize $Q(\mathbf{g})$, including ones based on greedy heuristics, spectral clustering, and sampling among other methods [61].

It has been found that many real networks divide quite naturally into communities [59], with values of high modularity and other related objective functions. This has a number of implications. For one, the communities in a network may represent functional sub-units of a system with high levels of internal communication and comparatively weak interdependence. Additionally, as with clustering, community structure impacts spreading processes on networks such as epidemics [62]. The existence of community structure in a network can also aid in data recovery tasks such as link prediction or network denoising [63], due to the information it gives about the likelihood of observing certain edges. Finally, the modules inferred by community detection may be of interest simply because they are likely to group similar nodes together, which aids in machine learning tasks such as topic modelling [64].

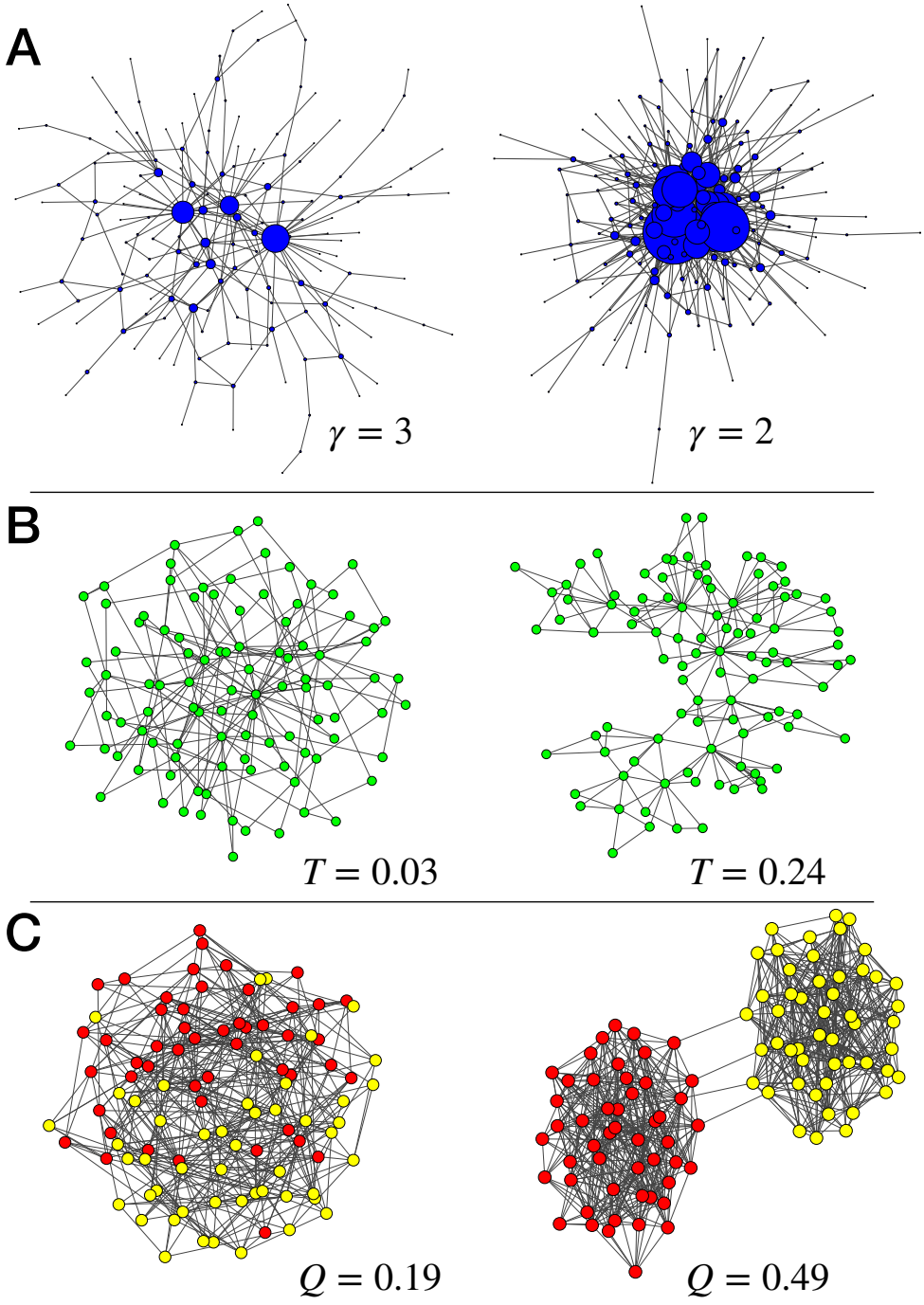

Fig. 1.2. Network properties illustrated with synthetic example graphs. **(A)** Graphs with power law degree distributions of different exponents γ, generated from the Chung-Lu model [65]. The size of a node is proportional to its degree. **(B)** Graphs with different transitivity values T, generated from the Holme-Kim model [66]. **(C)** Graphs with different modularities Q, generated from the stochastic block model [67]. Community assignments are indicated with red and yellow.

1.3. Statistical models of network structure

1.3.1. Graph models

Now that we have reviewed some key structural characteristics of real networks, we can begin to discuss how to model networks with these properties. As with mathematical modelling endeavors in other scientific disciplines, a critical element of most graph models is the presence of randomness. These graph models allow us to incorporate uncertainty into our description of a network's origins and predicted behavior, generate synthetic network data for analysis, and fit real network data with well understood statistical estimation procedures.

We have already seen the appearance of a graph model in Sec. 1.2.4, the configuration model, for motivating the modularity measure of community structure. In this case, the model serves the purpose of a *null model* to which we can compare the topology of a real network. However, there are numerous other choices of graph models one can use for this purpose [68–70], which preserve different characteristics of the network and can be used to determine whether or not a given structural feature in a real network is a by-product of fixing a different aspect of the network's structure. In the case of modularity, we want to see how much the connectivity within the network's assigned communities differs from the expected connectivity in a null model where degrees are fixed. As we will see, graph models are useful in many ways beyond just functioning as null models of network structure.

The configuration model is one of the simplest graph models, as it only requires that we specify the degree of each node. This allows all other aspects of the topology to vary completely at random subject to this constraint. However, we can specify even less information about the network by simply fixing the *average* degree $\langle k \rangle =$

$2m/n$. We can generate a network with a fixed number of nodes n and average degree $\langle k \rangle$ by placing edges completely at random between $m = \langle k \rangle n/2$ pairs of nodes i, j, of which there are $\binom{n}{2}$ in total. This called the *Erdős-Renyi model*, a cornerstone of random graph modelling since its creation in the late 1950's [71]. At the same time, a very similar model for networks was introduced by Gilbert [72] that instead fixes the *expected* number of edges by placing an edge independently at random in each of the $\binom{n}{2}$ possible locations with probability p. Both networks together are colloquially referred to as *random graphs*, and have served as the foundation for a wide variety of more sophisticated graph models. Though both models are equivalent in the limit $n \to \infty$, the latter model introduced by Gilbert is typically the version that is used in calculations as it permits simple analytical analysis [1, 8].

Combining aspects of the Gilbert model and the configuration model is the *Chung-Lu model* [65], which assigns a variable θ_i to each node i and generates a random graph with separate connection probabilities $\theta_i \theta_j / (\sum_i \theta_i)$ for each pair of nodes i, j. In expectation, we have $k_i = \theta_i$, and so we can specify $\boldsymbol{\theta}$ to match the degree distribution of a reference network when generating from the Chung-Lu model. The Chung-Lu model is generally more analytically tractable than the configuration model, and so it is preferred in most computations, just as the Gilbert model is preferred to the Erdős-Renyi model.

Despite their simple, elegant formulations and easy analytical treatment, random graphs and the configuration/Chung-Lu models are not very useful for most applications of graph modelling, as they do not capture the unique properties of real networks discussed in Sec. 1.2. For instance, since every edge in the random graph is placed independently with probability p, the probability P_k that a node in a random

graph has degree k is given by the binomial probability

$$P_k = \binom{n-1}{k} p^k (1-p)^{n-1-k}. \tag{1.14}$$

The degree thus follow a binomial distribution, which is tightly peaked about its mean value $p(n-1)$. On the other hand, as mentioned in Sec. 1.2.1, real networks tend to display highly heterogeneous (or heavy-tailed) degree distributions, and so the random graph model fails to capture this critical property. The configuration and Chung-Lu models are slightly better, as we specify the entire degree distribution (exactly or in expectation) as part of the model, but even these models have major drawbacks. For one, like the random graph, they produce networks with a vanishing transitivity (Eq. 1.10) as the network becomes large [1]. Since the clustered nature of real networks plays such a major role in their function, this shortcoming is quite significant.

There are countless alternative graph models that attempt to improve on the random graph and configuration/Chung-Lu models by capturing key structural features present in real networks. Two highly influential network models emerging in the late 1990's, the *Barabási-Albert* model [47] and the *Watts-Strogatz* model [42], laid the groundwork for many such models. The Barabási-Albert model—based on the Price model from decades earlier developed for directed citation networks [49]—aims to model the formation of the heavy-tailed (more specifically power law) degree distributions that the random graph fails to produce. In brief, this model posits that *preferential attachment* (also known as *cumulative advantage* or the *Matthew effect* [73]) causes the heterogeneous degree distributions we see in real networks through a network growth process where newly added nodes connect to existing nodes in proportion to their degree. On the other hand, the Watts-Strogatz model provides an

explanation for the simultaneously high levels of clustering and short path lengths seen in real networks, the first feature not found in either the random graph or the configuration/Chung-Lu models. In the Watts-Strogatz model, the network begins as a regular lattice (with a high degree of clustering but long path lengths), then edges are rewired at random to introduce shortcuts, which reduces the average path length but maintains a high level of clustering so long as only a small fraction of edges are rewired.

Common to both the Barabási-Albert and Watts-Strogatz models, however, is the lack of a mechanism explaining the existence of communities in networks. The foundational model addressing this issue is the *stochastic block model* (SBM) [67], which in its most common form takes as input a partition $\mathbf{g}$ of the nodes and generates a network by placing edges independently at random between nodes with a probability $\omega_{g_i g_j}$ that only depends on the group (or "block") assignments of the nodes. (We will analyze this model in more detail soon.) Like the Gilbert model of random graphs—which is just the SBM with a single community—the SBM permits simple analytical treatment, which has made it popular as a model to fit to real network data. This is in contrast with the Barabási-Albert and Watts-Strogatz models, which are largely used as network null models in practice. Due to their similar constructions, the standard SBM faces many of the same shortcomings as the random graph, and consequently there have been a number of proposed extensions to this model which make it more suitable for practical applications. These include generalizations with multiple group memberships for each node [74], hierarchical block structures [75], and weighted edges [76] among others. Perhaps the most significant extension to the SBM is the "degree-corrected" SBM (DCSBM) [77], which combines the Chung-Lu model and SBM by proposing connection probabilities proportional to $\theta_i \theta_j \omega_{g_i g_j}$, with

a vector of parameters $\boldsymbol{\theta}$ that control the degrees of the nodes.

With the advent of a broad array of network models in the 2000's and 2010's, largely inspired by the models we have described, the focus of the theoretical network science community in recent years has shifted to the modelling of more sophisticated and flexible representations of the structure and dynamics of networked systems—including temporal networks, multilayer networks, and hypergraphs, among others—-as well as the problem of *model estimation*, or fitting network models to data. We describe in the next section the Bayesian paradigm for inference with network data, a powerful framework that is the basis for much of the modern research in statistical network estimation.

1.3.2. Bayesian inference of community structure

Bayesian statistics is an extremely broad area with methods for addressing problems in regression, interval estimation, clustering, data imputation, density estimation, and many other statistical tasks [78]. Common to all these methods is that uncertainties in inferences are quantified using probabilities that express belief about an event, and these probabilities are updated from an initial prior belief upon observing data. Here we discuss the specific application of Bayesian inference to community detection, but note that there are a number of other network inference tasks for which Bayesian inference provides a principled, robust solution [79–83].

Assume we have an observed network G consisting of n nodes, and we would like to find a division of these nodes into disjoint communities. As before, we represent a community division with a length n vector $\mathbf{g}$ such that g_i is the label of the community to which i belongs. In the Bayesian framework, we assume that G was created

using some statistical generative model $P(G|\mathbf{g}, \boldsymbol{\theta})$, which depends on an underlying partition $\mathbf{g}$ and other model parameters $\boldsymbol{\theta}$. For example, if we assume that G was generated from an SBM with some partition $\mathbf{g}$, then the parameters $\boldsymbol{\theta}$ are the matrix of connection probabilities $\boldsymbol{\omega}$, such that $\omega_{g_i g_j}$ is the probability of placing an edge between nodes i and j. $P(G|\mathbf{g}, \boldsymbol{\theta})$ is called the *model likelihood* in the context of inference. Using Bayes' rule, we can then find the *posterior distribution*

$$P(\mathbf{g}, \boldsymbol{\theta}|G) = \frac{P(G|\mathbf{g}, \boldsymbol{\theta})P(\mathbf{g}, \boldsymbol{\theta})}{P(G)}. \tag{1.15}$$

$P(\mathbf{g}, \boldsymbol{\theta})$ is a *prior* distribution that quantifies our beliefs about what values $\mathbf{g}$ and $\boldsymbol{\theta}$ may take, before observing G. In many cases, we take this to be an uninformative distribution with high variance, to assume little information about the partition and model parameters. However, in other circumstances a stronger prior is necessary either to avoid issues during the model fit or to constrain the posterior to be consistent with domain expertise. The denominator,

$$P(G) = \sum_{\mathbf{g}} \int P(G|\mathbf{g}, \boldsymbol{\theta})P(\mathbf{g}, \boldsymbol{\theta})d\boldsymbol{\theta}, \tag{1.16}$$

is a normalization constant called the *model evidence*. The posterior distribution (Eq. 1.15) is the fundamental object of interest in Bayesian inference, as it quantifies our degree of belief about each possible configuration of the model variables $\mathbf{g}, \boldsymbol{\theta}$, taking into account both our prior beliefs (through $P(\mathbf{g}, \boldsymbol{\theta})$) and the data (through $P(G|\mathbf{g}, \boldsymbol{\theta})$). By having a full probability distribution over these variables at our disposal, we can construct estimates (with uncertainties) of these quantities, compute expectation values of functions of these variables, or draw samples.

For many network models and prior choices, $P(G|\mathbf{g}, \boldsymbol{\theta})$ and $P(\mathbf{g}, \boldsymbol{\theta})$ take tractable analytical forms. (Notably, network models that represent a sequential growth process, similar to the Barabási-Albert model, do not in general have trivially tractable likelihoods, as we must account for the order in which the nodes arrived which is not available with a single static network. Efficient computational approximations and analytical results for these types of models have become of increased interest [84, 85].) The crucial component of the posterior distribution that makes Bayesian inference difficult is the model evidence (Eq. 1.16), as we typically cannot perform the integrations/summations necessary to evaluate the expression. Even using numerical methods, the dimensionalities of these integrations/summations are usually too high to compute $P(G)$ to any reasonable degree of accuracy. Not all hope is lost, however, as there is a suite of mathematical and computational tools that have been developed to tackle precisely the problem of estimating the posterior distribution in Eq. 1.15. (See Appendix A for details on a few common techniques and an example calculation.) These methods will return either point estimates or samples of $\mathbf{g}$ and $\boldsymbol{\theta}$, which can be further examined to gain an understanding of the community structure of the network.

1.4. Contributions of this thesis

In this chapter, I briefly reviewed key definitions, measures, and models for networks, which will provide a foundation for the research I present in the upcoming chapters. In the remainder of the thesis, I will discuss my work developing novel measures and methods for network analysis, which are aimed at addressing two primary objectives: to characterize structure in networks with metadata (Chapters 2 and

3), and to develop scalable, accurate, and interpretable inference techniques for real-world network data (Chapters 4 and 5).

In Chapter 2, I present my work on signed networks, in which edges can be either positive (friendship, trust, alliance) or negative (dislike, distrust, conflict). Early literature in graph theory theorized that such networks should display "structural balance," meaning that certain configurations of positive and negative edges are favored and others are disfavored. I propose two measures of balance in signed networks based on the established notions of weak and strong balance, and compare their performance on a range of tasks with each other and with previously proposed measures. In particular, I ask whether real-world signed networks are significantly balanced by these measures compared to an appropriate null model, finding that indeed they are, by all the measures studied. I also test the ability to predict unknown signs in otherwise known networks by maximizing balance. In a series of cross-validation tests I find that these measures are able to predict signs substantially better than chance. This chapter is based on work published with George Cantwell and Mark Newman in *Physical Review E* [86].

In Chapter 3, I return to the topic of incorporating metadata into measures of network structure, this time in the context of spatial networks of socioeconomic data. To mitigate issues associated with traditional spatial measures of inequality and segregation, I develop an information theoretic approach based on the generalized Jensen-Shannon divergence for quantifying variation in distributions of socioeconomic attributes across spatial networks of adjacent regions. I apply my methodology in a series of experiments to study the network of neighboring census tracts in the continental US, quantifying the decay in two-point distributional correlations across the network, examining the county-level socioeconomic disparities induced from the ag-

gregation of tracts, and constructing an algorithm for the division of a city into homogeneous clusters. This chapter is based on work published in a single-author paper in *Physical Review Research* [87].

In Chapter 4, I switch gears and discuss methods for statistical inference with network data, addressing the serious shortcoming of belief propagation (see Appendix A) performing poorly in the common case of networks that contain short loops. Here, I provide a solution to this long-standing problem, deriving a belief propagation method that allows for fast calculation of probability distributions in systems with short loops, potentially with high density, as well as giving expressions for the entropy and partition function, which are notoriously difficult quantities to compute. Using the Ising model as an example, I show that this approach gives excellent results on both real and synthetic networks, improving substantially on standard message passing methods. I also discuss potential applications of this method to a variety of other problems. This chapter is based on work published with George Cantwell and Mark Newman in *Science Advances* [88].

In Chapter 5, I continue with this inference theme by presenting a method for identifying representative community divisions of networks. Techniques for detecting community structure in networks typically aim to identify a single best partition of network nodes into communities, often by optimizing an objective function such as modularity. However, in real-world applications there are typically many competitive partitions with objective scores close to that of the global optimum and the true community structure is more properly represented by an entire set of high-scoring partitions than by just the single optimum. Such a set can be difficult to interpret since its size can easily run to hundreds or thousands of partitions. In this chapter I present a solution to this problem in the form of an efficient method that clusters

similar partitions into groups and then identifies an archetypal partition as a representative of each group. The result is a succinct, human-readable summary of the form and variety of community structure in any network. I demonstrate the method on a range of example networks. This chapter is based on work with Mark Newman that is currently under review [89].

In Chapter 6, I conclude the thesis by reflecting on this work and discussing avenues for future research in these areas.

Chapter 2.

Balance in Signed Networks

As discussed in Sec. 1.1, a network in its simplest form consists of a collection of nodes joined together in pairs by edges, but many networks have additional features as well. The first primary goal of this thesis is to characterize structure in networks with metadata, and in this chapter we consider one case of particular interest, that of *signed networks*, meaning networks in which the edges are either positive or negative [1, 90, 91]. The most common example is a social network that represents patterns of both amity and enmity among a group of individuals: positive edges represent friendship, negative ones animosity. As we will see, quantifying the interplay of the topology and edge sign metadata in these networks can provide us with a wealth of insight into the nature of conflict and resolution in social systems, as well as techniques for inferring missing data.

2.1. Introduction

Studies of signed networks go back at least to the classic work of Harary in the 1950s, who argued, largely on formal rather than empirical grounds, that certain patterns of signs should be more common than others—the enemy of my enemy should be my friend, for example [90]. Networks that display such regularities are said to be *structurally balanced*, or just *balanced* for short. A natural question to ask is whether real signed networks are in fact balanced. Despite a considerable amount of research on this issue, however, the jury is still out. Some researchers have claimed that real networks are balanced, at least partially, while others have claimed that they are not [92–94].

There are two primary reasons for the disagreement. First, there is more than one proposed definition of structural balance in networks. Cartwright and Harary [95] proposed that a network is balanced if all closed loops in the network contain an even number of negative edges. This condition, which we will refer to as *strong balance*, is a stringent one that is rarely if ever completely satisfied in real networks. As we will see, however, one can define measures of partial balance that quantify how close a network comes to Cartwright and Harary's ideal.

Strong balance is an attractive formulation in part because of a theorem due to Harary [90], which says that any network displaying perfect strong balance is *clusterable*, meaning its nodes can be divided into some number of disjoint sets such that all edges within sets are positive and all edges between sets are negative. Thus strong balance provides a possible theoretical basis for insularity or cliquishness in social networks: if networks naturally display strong balance, then they also naturally divide into communities such that people like members of their own community and

dislike members of other communities.

While strong balance is a sufficient condition for clusterability, however, it turns out that it is not a necessary one, as shown by Davis [96], who demonstrated that for a network to be clusterable in the sense above, one requires only a lesser form of structural balance, namely that there be no closed loops in the network with exactly one negative edge. We will refer to this condition as *weak balance*. Weakly balanced networks are a superset of strongly balanced ones—every strongly balanced network is necessarily also weakly balanced—but weak balance alone is enough to explain insularity in networks and division into antagonistic communities.

Alternatively, causality might run in the opposite direction: if a population is intrinsically divided into two or more antagonistic factions—Montagues and Capulets, Roundheads and Cavaliers, Hatfields and McCoys—then by definition the resulting network will be balanced. Indeed, if there are exactly two factions then the network will be strongly balanced, since every closed loop must traverse negative edges between the factions an even number of times. If there are three or more factions then the network will, in general, be only weakly balanced.

Thus we have two competing notions of what it means for a network to be balanced. It is in part the lack of consensus about which of the two to adopt that makes it hard to say whether real networks are in fact balanced or not.

The second reason for the lack of agreement is that in order to say whether a network is balanced we need to specify the scale on which balance is to be assessed. Even if we can agree on a measure of balance, how do we know whether the observed level is high or low? A natural approach is to compare the level to what we would expect on the basis of chance, i.e., to the level in some kind of null model, but it is by no means universally agreed what form such a null model should take.

In this chapter we do several things. First, we consider a number of possible measures of both strong and weak balance. Some of the measures we discuss have been proposed previously; some we propose here for the first time. Second, we consider possible null models against which to compare levels of balance, choosing one we believe to be appropriate for the questions we are interested in. Third, we use our measures and our null model to quantify structural balance in real-world signed networks, finding that the networks we consider are indeed significantly more balanced, at least according to our measures, than we would expect on the basis of chance.

The presence of structural balance in networks is interesting in its own right, for the hints it gives us about the growth and function of social networks. But we can also use our knowledge of balance to perform other tasks. As an example, we demonstrate how it can be used to make predictions of the signs of unobserved edges. By simply assigning edges the choice of sign that makes the overall network most balanced, we show that we can predict the correct value of missing edge signs in test networks substantially better than chance. As a corollary, this also gives us some insight about which are the best measures of balance: all of the measures we consider perform well in the sign prediction task, but the measure based the weak notion of balance appears to perform somewhat better, perhaps indicating that weak balance is a better description of the behavior of real-world networks than strong balance.

There has been a significant amount of previous work to define and study structural balance in signed networks [97], including methods and metrics motivated by spin glasses [92, 98–100] and dynamical systems [101, 102], spectral methods [103–105], and Harary's "line index" of imbalance [106], as well as walk-based approaches [94, 107–110], of which our own proposed methods can be considered an example. Rather than giving a comprehensive review of all of these approaches, we focus here primar-

ily on the walk-based approaches, several of which share features with our methods [94, 108–110], although there are some crucial differences as well. Perhaps the approach most similar to ours is that of Singh and Adhikari [110], who propose a measure of balance motivated by the notion of strong balance that accounts for the lesser effect of long loops on social tension. We propose two similar measures, one for strong balance and one for weak, though with a different choice of weighting for short and long loops. Another important difference between our work and that of Singh and Adhikari lies in the choice of null model, for which they use ensembles of networks where positive, negative, and non-edges are placed randomly. By contrast, in our work we randomize only the signs of the edges and not their positions, which we argue is essential for proper quantification of statistically significant balance in networks.

2.2. Methods

Real-world signed networks are rarely, if ever, perfectly balanced, so to study balance in such networks we need a way to quantify exactly how balanced they are. Following previous authors, we consider measures that quantify the number of closed loops in a network that violate either the strong or the weak notion of balance, meaning respectively that they have either an odd number of negative edges (strong balance) or exactly one negative edge (weak balance).

This alone, however, is not enough to define a practical measure because of another feature of networks, that the number of closed loops of a given length increases rapidly with length. If one were simply to count closed loops, the count would be dominated by the longest loops in the network solely because they are more numer-

ous. It seems unlikely, however, that long loops play much of a role in real-world issues of balance. Few people really care if a friend of a friend of a friend is an enemy or not. Realistically, we expect that it is the short loops, not the long ones, that dominate network balance. The second defining feature of the measures we consider, therefore, is that they weight short loops more heavily than long ones.

2.2.1. Balance measures

Consider an undirected signed graph or network G. As defined in Sec. 1.2.2, a closed walk in a such network is any path that begins and ends at the same node, and a simple cycle is a closed walk that does not visit any node twice, other than the start/end node, which is visited exactly twice. The strong definition of balance then says that G is a balanced network if, and only if, every simple cycle in G has an even number of negative signs. The weak definition of balance, by contrast, says that a network is balanced if, and only if, it contains no simple cycles with exactly one negative edge (meaning that any other number is fine). We can also say that individual cycles are strongly or weakly balanced by the same criteria.

We can use these ideas to define a measure $B(z)$ of the level of imbalance in a network thus:

$$B(z) = \sum_{k=1}^{\infty} \frac{I_k}{z^k}, \tag{2.1}$$

where I_k is the number of imbalanced simple cycles of length k and $z > 1$ is a free parameter. This measure takes the form of a weighted count of imbalanced cycles in which longer cycles get downweighted by a geometric factor z^k. Note that the sum in (2.1) could in principle start at $k = 2$ without changing the value of $B(z)$, since there are no cycles of length one, but it will be convenient for subsequent developments to

start at $k = 1$.

We can define a measure of this type for either the weak or strong notion of balance. Let us look first at the weak version, meaning that I_k will be the number of simple cycles of length k that contain exactly one negative edge. An immediate problem we encounter with applying this measure is the difficulty of making practical estimates of the number of simple cycles of a given length in an arbitrary network. There is no elementary analytic approach for counting cycles, and numerical methods are hampered by the very rapid increase of I_k with k, which makes exhaustive enumeration of cycles possible only for small k and small networks. Instead, therefore, we approximate the number of simple cycles by the number of closed walks, which is relatively straightforward to compute. To count the number of weakly imbalanced closed walks of length k, we remove all the negatives edges from the network and then look at the number of walks of length $k - 1$ between the (former) endpoints of those edges. Reinserting the negative edges again then closes the walks, creating loops of length exactly k, each with exactly one negative edge.

Substituting closed walks for simple cycles is a good approximation when the cycles are short. Indeed, for cycles of length three it is exact: closed walks and simple cycles are the same thing for length three. As the length increases the approximation gets worse [111], but in practice this may not matter very much. The imbalance metric of Eq. (2.1) discounts long loops, so the fact that our count is only approximate may not make much difference.

To put the developments in mathematical terms, let us denote the structure of our network by two $n \times n$ adjacency matrices $\mathbf{P}$ and $\mathbf{N}$, for the positive and negative edges respectively. Thus, matrix $\mathbf{P}$ has elements $P_{ij} = 1$ if nodes i and j are connected by a positive edge and 0 otherwise, and similarly $N_{ij} = 1$ if i and j are connected

by a negative edge and 0 otherwise. Then our imbalance measure, which we will denote $B_W(z)$ with subscript W to indicate weak balance, is given by

$$B_W(z) = \tfrac{1}{2} \sum_{ij} N_{ij} \sum_{k=1}^{\infty} \frac{1}{z^k} \big[\mathbf{P}^{k-1}\big]_{ji} = \tfrac{1}{2} \operatorname{Tr}\big[\mathbf{N}(z\mathbf{I} - \mathbf{P})^{-1}\big], \tag{2.2}$$

the factor of $\frac{1}{2}$ compensating for the fact that the sum counts each loop twice, once in each direction.

In fact, it will be convenient to introduce a rescaled parameter $\alpha = z/\lambda_P$, where λ_P is the leading (most positive) eigenvalue of $\mathbf{P}$. For $\alpha > 1$ this ensures that the sum in (2.2) will converge, and we can write

$$B_W(\alpha) = \tfrac{1}{2} \operatorname{Tr}\big[\mathbf{N}(\alpha\lambda_P\mathbf{I} - \mathbf{P})^{-1}\big]. \tag{2.3}$$

Another way to interpret the parameter α is to write $\alpha^{-k} = \mathrm{e}^{-k/k_0}$, where $k_0 = 1/\ln\alpha$ is a "decay length" that determines the length scale on which the contributions from longer walks are discounted. Thus, for example, if we choose $\alpha = 2$, we have $k_0 = 1/\ln 2 \simeq 1.44\ldots$, and three such decay lengths give us a 95% decay at distance a little greater than 4.

An analogous measure $B_S(\alpha)$ can be defined for the strong notion of balance. Again we approximate the number of imbalanced simple cycles by the number of closed walks, which we can calculate as follows. Consider the matrix $\mathbf{P} - \mathbf{N}$, which has elements $+1$ for positive edges, -1 for negative edges, and 0 otherwise. The kth power of this matrix counts walks of length k, times $+1$ if they contain an even number of minus signs and -1 if odd. Thus the diagonal term $[(\mathbf{P} - \mathbf{N})^k]_{ii}$ is equal to the number of balanced closed walks starting and ending at node i minus the number of

imbalanced ones. Summing over all i, we then have [111]

$$B_k - I_k = \frac{1}{2k}\,\mathrm{Tr}\big[(\mathbf{P}-\mathbf{N})^k\big], \tag{2.4}$$

where B_k and I_k are the total number of balanced and imbalanced closed walks. The initial factor of $\frac{1}{2}$ again compensates for the fact that we count each loop in both directions, and the factor of $1/k$ compensates for the fact that each loop is counted repeatedly starting from each of the k points along its length.

Conversely, consider the matrix $\mathbf{P}+\mathbf{N}$, which is simply the adjacency matrix of the complete network, ignoring signs—every edge, positive or negative, is represent by a $+1$ in this matrix. The total number of closed walks of length k, both balanced and imbalanced, is given by

$$B_k + I_k = \frac{1}{2k}\,\mathrm{Tr}\big[(\mathbf{P}+\mathbf{N})^k\big]. \tag{2.5}$$

Subtracting (2.4) from (2.5) and dividing by 2, we get an expression for the number of imbalanced loops:

$$I_k = \frac{1}{4k}\,\mathrm{Tr}\big[(\mathbf{P}+\mathbf{N})^k\big] - \frac{1}{4k}\,\mathrm{Tr}\big[(\mathbf{P}-\mathbf{N})^k\big]. \tag{2.6}$$

Substituting this into Eq. (2.1) then gives us our measure of strong imbalance:

$$B_S(z) = \tfrac{1}{4}\sum_{k=1}^{\infty}\frac{1}{kz^k}\,\mathrm{Tr}\big[(\mathbf{P}+\mathbf{N})^k\big] - \tfrac{1}{4}\sum_{k=1}^{\infty}\frac{1}{kz^k}\,\mathrm{Tr}\big[(\mathbf{P}-\mathbf{N})^k\big]. \tag{2.7}$$

Making use of the matrix identity

$$\sum_{k=1}^{\infty}\frac{\mathrm{Tr}\,\mathbf{M}^k}{k} = -\log\det(\mathbf{I}-\mathbf{M}), \tag{2.8}$$

this can also be written as

$$B_S(z) = \tfrac{1}{4} \log \frac{\det[z\mathbf{I} - (\mathbf{P} - \mathbf{N})]}{\det[z\mathbf{I} - (\mathbf{P} + \mathbf{N})]}, \tag{2.9}$$

which is valid whenever the sums in (2.7) converge. As with $B_W(z)$ it is convenient to reparametrize this expression in terms of $\alpha = z/\lambda^*$, where λ^* is the larger of the leading eigenvalues of $\mathbf{P} + \mathbf{N}$ and $\mathbf{P} - \mathbf{N}$, so that

$$B_S(z) = \tfrac{1}{4} \log \frac{\det[\alpha\lambda^*\mathbf{I} - (\mathbf{P} - \mathbf{N})]}{\det[\alpha\lambda^*\mathbf{I} - (\mathbf{P} + \mathbf{N})]}, \tag{2.10}$$

which ensures convergence of the sums when $\alpha > 1$.

2.2.2. Previous measures of network balance

A number of previous researchers have also proposed measures of structural balance in networks. Estrada and Benzi [94] (henceforth EB) define a measure

$$B_{\text{EB}} = \frac{1 - K}{1 + K}, \tag{2.11}$$

where

$$K = \frac{\sum_k \text{Tr}\big[(\mathbf{P} - \mathbf{N})^k\big]/k!}{\sum_k \text{Tr}\big[(\mathbf{P} + \mathbf{N})^k\big]/k!}. \tag{2.12}$$

The quantity K is in some ways analogous to our measure of strong imbalance, Eq. (2.9), but it downweights longer loops by a larger factor $1/k!$, compared to the geometric factor $1/z^k$ that we employ. This results in some elegant mathematical expressions but has the disadvantage that there is no way to set the length scale on which loops are discounted. EB also define their measure not by K itself but by the formula (2.11),

which can be interpreted as a ratio of weighted counts of unbalanced and balanced loops.

Singh and Adhikari [110] (henceforth SA), in considering the measure of EB, object to the weight factor $1/k!$ and propose instead to use a geometric factor as we do, defining a measure

$$B_{\mathrm{SA}}(z) = \frac{\sum_k \mathrm{Tr}\big[(\mathbf{P}-\mathbf{N})^k\big]/z^k}{\sum_k \mathrm{Tr}\big[(\mathbf{P}+\mathbf{N})^k\big]/z^k}. \tag{2.13}$$

This is again somewhat analogous to Eq. (2.9), though it is not directly based on the actual number of imbalanced loops, and moreover appears to neglect the factor of $1/k$ that accounts for the k possible starting points around a loop of length k.

We compare the performance of the four measures discussed here, our own measures B_W and B_S and the measures of EB and SA, on a number of problems concerning balance in networks.

2.2.3. Null models

As discussed in Sec 2.1, measures of imbalance are difficult to employ on their own because we lack a scale on which to calibrate their values. If we calculate a value of, say, $B_W = 0.5$ for a particular network how do we know if that value is large or small? One way to answer this question is to compare our numbers with values calculated in an appropriate null model.

The broader question we are addressing in when calculating measures of balance is whether the arrangement of positive and negative edges within a network is somehow special, different from what we would expect on the basis of chance. Since our focus is on the arrangement of signs within the larger network, and not on the arrangement of edges per se, the natural null model to consider is one in which the signs in a network

are randomized while keeping the locations of the edges fixed. In the particular null model we consider here, we also keep the overall number of positive and negative signs fixed, to make the randomized networks more directly comparable with the original.

This null model or ones similar to it have been used in a number of previous works [112–114], but it is not the only possible choice [110, 115]. Singh and Adhikari [110], for example, employ a null model in which both the signs and the positions of the edges are randomized. This results in networks whose structure, in terms of edge placement, is very different from that of the original network, which makes it difficult to know how much of any observed difference in balance is due to the pattern of signs and how much to the edge positions. One could also consider a model in which the edge positions are randomized but the signs on the edges are fixed, although this suffers from the same problems as the model of Singh and Adhikari. The null model we employ avoids these difficulties by randomizing the signs only.

Arguably, in many real-world situations—coworkers in an office, for instance, or children in a school class—one indeed has no choice about who one interacts with, so that the positions of the network edges are fixed. The only degree of freedom is the nature of the interactions, whether they will be friendly or antagonistic. A model that fixes the edge positions but varies their signs is thus a natural choice in such cases.

2.3. Results

As examples of the techniques introduced here, we consider their application to two data sets, one from the field of international relations, representing positive and negative ties between countries [116], and the other from sociology, representing ties

between a group of university freshmen [117]. For both data sets we use our measures to quantify structural balance, and for the international relations data we also test our ability to make predictions of the signs of unobserved edges.

The international relations data set contains many details of inter-country interactions over a period of several decades, but here we focus on two aspects in particular: alliances and wars. We construct a set of signed networks, one for each year in the 70-year period from 1938–2008, in which nodes represent countries and two countries are connected by a positive tie if they have a formal alliance in that year and a negative tie if there is a militarized dispute between them. In the rare cases in which countries have both an alliance and a war in the same year we take the corresponding edge to be negative. (The same methodology was used previously in [118].) Only countries for which we have data are included in our networks. The number of nodes ranges from 25 to 155 with a median of 105, and the number of edges ranges from 46 to 1230 with a median of 615. The signs of the edges are predominantly positive—most countries have good relations. The fraction of negative edges ranges from 1.8% to 45.1% with a median of 5.5%. (The outliers with the largest number of negative edges all fall during the Second World War. The median fraction of negative edges between 1940 and 1945 was 44%.)

The university freshman data set describes relationships between a group of first-year students, all at the same university, and consists of networks collected at seven different time points. At each time point the students were asked to rate their relationships with all other students in the group on a five-point scale of (1) "best friend", (2) "friendship", (3) "friendly relationship", (4) "neutral relationship", or (5) "troubled relationship". Students could also say they did not know the person in question. Further discussion of the scale can be found in [117]. We construct a set of signed

networks, one for each time point, in which two students are connected by a positive tie if each rates the other as a 3 or lower, and a negative tie if one or both rates the other as a 5. Neutral relationships are not represented in the network, which means that there is no difference in our representation between having a neutral relationship and having no relationship at all. While this is not ideal, it seems like the best strategy given that there is no principled way to decide whether a neutral edge should be considered positive or negative. Of the seven networks constructed in this way, we discard three because of sparse or missing data, leaving four that we analyze here. The number of nodes in the networks is 34 at all time points and the number of edges ranges from 174 to 227 with a median of 225.5. The fraction of negative edges ranges from 12% to 14% with a median of 13%.

2.3.1. Balance relative to the null model

To quantify the level of balance in a network, we compute the ratio between the value B of each metric and the average value $\langle B \rangle$ of the same metric on a selection of randomized networks drawn from the null model described in Section 2.2.3:

$$\eta = \frac{B}{\langle B \rangle}. \tag{2.14}$$

Figure 2.1 shows the values of this ratio as a function of time for the international relations networks for the four balance metrics considered in this chapter, along with the mean for the null model and an indication of the fluctuation of the results about that mean (the bands shown are two standard deviations). As the figure shows, in each case actual imbalance values, for all measures, are far below what would be expected for the null model. (An alternative way to represent the same results would

be to compute a z-score, but we prefer the representation of Fig. 2.1 since it shows explicitly the size of the fluctuations in the null-model values.) Figure 2.2 shows results from the same experiment performed on the university freshman networks.

For this calculation the metrics B_W and B_S are both computed with a parameter value $\alpha = 2$, as discussed on Section 2.2.1, and we use the corresponding value for the parameter in the metric of Singh and Adhikari (SA) [110] as well. (The metric of Estrada and Benzi (EB) [94] has no free parameters.) We have also experimented with a range of alternative parameter values, but find that the results do not depend strongly on our choice.

Our goals here are two-fold. First, we wish to see if real networks are indeed unusually balanced relative to an appropriate null model. Second, if they are balanced in this sense, we wish to see which of our notions of balance most clearly distinguishes real networks from their null model counterparts. As Fig. 2.1 shows, all four metrics give extremely low η values relative to the null model, all of which would be statistically significant at the $p < 0.05$ level in all years if we assume a normal distribution within the null model. The most significant values occur during the World War II period, specifically between 1940 and 1945, and this effect is especially pronounced for the three metrics based on the strong notion of balance. As mentioned in the introduction, strong balance is expected in cases where a network is divided into just two main factions, which was the case during World War II. Note that, during this period, η is actually *greater* than in other periods, but that the values for the null model have a much lower standard deviation than in other years, making the results for the real networks more statistically significant relative to the null model.

Figure 2.2 for the university networks shows similar behavior, although the η values are less extreme than those for the international relations networks. This might be due

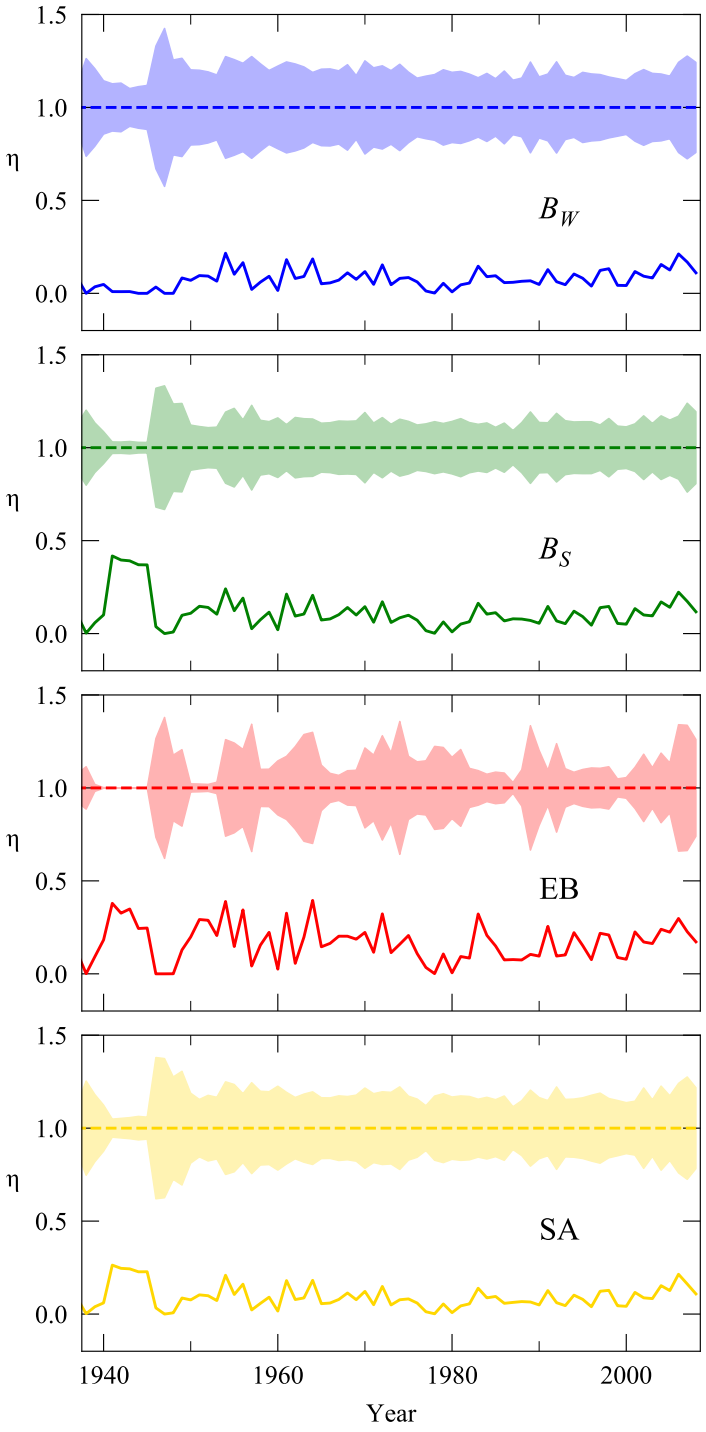

Fig. 2.1. Level of imbalance in the international relations networks for 1938–2008, as measured by the ratio η defined in Eq. (2.14), for each of the four balance measures studied here. The dotted lines indicate the null model mean, which falls at $\eta = 1$ by definition, and the surrounding bands denote two standard deviations of the fluctuations around this mean. The solid lines represent the values for the observed networks. All networks are significantly less imbalanced than the null model by all four measures.

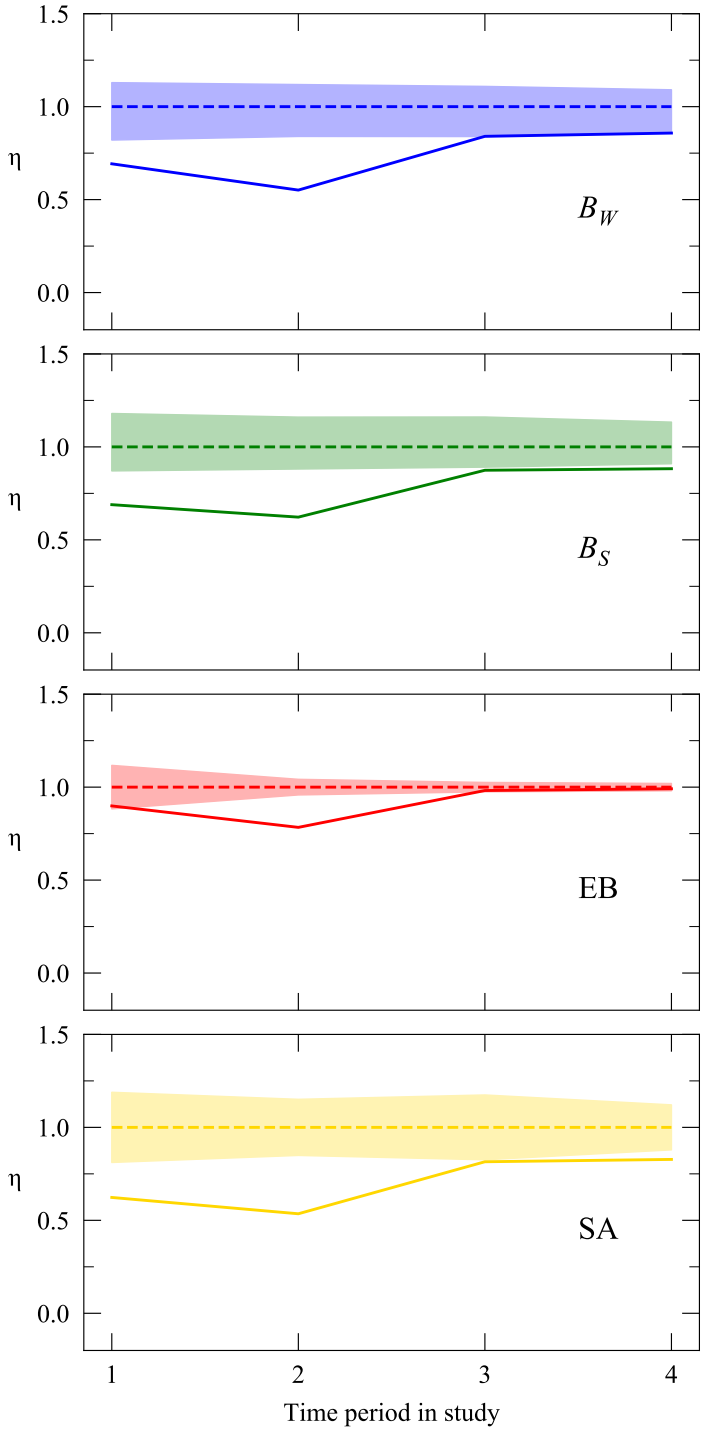

Fig. 2.2. Levels of imbalance in the university freshman networks. The dotted lines indicate the null model mean and the surrounding bands denote two standard deviations of the fluctuations around this mean. The solid lines represent the values for the observed networks.

to a lower level of factionalism for the students than for international relations, or to measurement error, or a combination of both.

The university data set also lends itself to being represented as a weighted, directed network, and one could consider generalizations of the methods presented here to such networks, although this is outside the scope of the present work.

2.3.2. Sign prediction

Consider a situation in which we know the positions of the edges in a signed network, but we know the signs of only some of the edges. The signs of the remaining edges are missing from our data, perhaps because they were not measured or recorded, or because our measurements are unreliable. Guha *et al.* [112], in studies of trust in online communities, suggested that it should be possible, using the patterns of known signs, to make predictions about the unknown ones, and in recent years a number of authors have developed algorithms to do this [105, 108–110, 119, 120]. (Looking for correlations between signs is not the only way to perform prediction—there are a whole range of network reconstruction algorithms that could be adapted for signed networks [121]—but our focus here is specifically on the use of known signs to predict unknown ones.)

A natural approach is to start from the assumption that the network is balanced [110, 119]. Consider the simple case where a sign is missing from just a single edge in the network and our goal is to guess the value of that sign given all the others. We assume that the best guess for the missing sign is the one that will make the network most balanced. This leaves open the question of which metric we should use to quantify balance, which we address by performing a cross-validation study in which we

artificially remove one sign from an otherwise complete network, then attempt to predict its value using each of our metrics in turn. Repeating this process for every edge in the network, we measure the average success of our predictions for each metric.

This "single edge" prediction test is arguably unrealistic—in most real-world scenarios there will be more than one sign missing from any incomplete data set, a point that we discuss further in Section 2.3.3. It is, nonetheless, a good starting point by virtue of being relatively computationally tractable for networks of the size considered here, which typically have a few hundred edges. We can just calculate directly the value of each of our balance metrics for the two possible choices of each sign and take the choice that gives the higher balance.

For larger network sizes this brute-force approach becomes more computationally demanding, but with a little ingenuity the calculation can still be done. The calculation of our metrics B_W and B_S relies on the computation of either a matrix inverse (for B_W) or a matrix determinant (for B_S), and there exist formulas that allow one to quickly recalculate inverses and determinants when only a few elements of a matrix are altered, as in this case. Consider, for instance, the weak balance measure B_W defined in Eq. (2.2). The primary computational task in evaluating this measure is the calculation of the *resolvent matrix* $\mathbf{R} = (z\mathbf{I} - \mathbf{P})^{-1}$. We can speed up this calculation as follows. First, we directly compute $\mathbf{R}$ for the original network and use it to evaluate B_W. This is a relatively slow operation: computing the inverse of an $n \times n$ matrix takes $\mathrm{O}(n^3)$ time in a naive implementation, and modestly better in more complex schemes. Then we consider in turn each edge in the network and compute the value of B_W when the sign of that edge is reversed. Reversing the sign of an edge between nodes i and j alters the values of P_{ij} and P_{ji} by ± 1, a change that we can write in the

low-rank form

$$\mathbf{P}' = \mathbf{P} \pm \mathbf{U}\mathbf{V}, \tag{2.15}$$

where $\mathbf{U}$ is an $n \times 2$ matrix with all elements zero except $U_{i1} = U_{j2} = 1$, and $\mathbf{V}$ is a $2 \times n$ matrix with all elements zero except $V_{1j} = V_{2i} = 1$. Then the *Woodbury matrix identity* [122] tells us that the new value of the resolvent $\mathbf{R}' = (z\mathbf{I} - \mathbf{P}')^{-1}$ is given by

$$\mathbf{R}' = \mathbf{R} \pm \mathbf{R}\mathbf{U}(\mathbf{I} \mp \mathbf{V}\mathbf{R}\mathbf{U})^{-1}\mathbf{V}\mathbf{R}, \tag{2.16}$$

which requires only the trivial inversion of the 2×2 matrix inside the brackets. Evaluation of the matrix products $\mathbf{R}\mathbf{U}$ and $\mathbf{V}\mathbf{R}$ and evaluation of the n^2 elements of $\mathbf{R}'$ all take $\mathrm{O}(n^2)$ time, so the running time to calculate the new value of B_W is also $\mathrm{O}(n^2)$, a substantial improvement on the $\mathrm{O}(n^3)$ time needed to calculate it from scratch.

Similarly for the strong balance measure B_S it is possible to evaluate the measure rapidly upon the change of single sign. This measure, defined in Eq. (2.9), involves the calculation of the determinant of the matrix $\mathbf{A} = z\mathbf{I} - (\mathbf{P} - \mathbf{N})$, whose value changes upon the flip of a sign to

$$\mathbf{A}' = \mathbf{A} \pm 2\mathbf{U}\mathbf{V}, \tag{2.17}$$

where $\mathbf{U}$ and $\mathbf{V}$ are as previously defined. (The determinant in the denominator of Eq. (2.9) does not change when a sign is flipped, so there is no need to recalculate it.) Then the *matrix determinant lemma* [123] states that the new value of the determinant is related to the old one by

$$\det(\mathbf{A} \pm 2\mathbf{U}\mathbf{V}) = \det(\mathbf{A}) \det(\mathbf{I} \pm 2\mathbf{V}\mathbf{A}^{-1}\mathbf{U}). \tag{2.18}$$

Once one has the inverse $\mathbf{A}^{-1}$ this computation can be performed quickly. The 2×2 matrix $\mathbf{I} \pm 2\mathbf{V}\mathbf{A}^{-1}\mathbf{U}$ can be calculated in time $\mathrm{O}(n^2)$ and its determinant in constant time, so again the overall calculation takes $\mathrm{O}(n^2)$ time. By contrast, calculating the determinant directly from scratch takes $\mathrm{O}(n^3)$ time (or slightly better using the fastest algorithms), so again we have a substantial improvement in speed over the direct calculation. For the other balance metrics considered here (EB and SA) there are similar shortcuts that can speed up calculations for larger networks, although we will not use them here.

Figure 2.3 shows the results of single-sign prediction calculations for our international relations networks as a function of time, for each of our four measures of balance. The vertical axis in the figure measures the fraction of all signs predicted correctly, also known as the *accuracy*. By contrast with the results shown in Figs. 2.1 and 2.2, performance on this task clearly varies between the different balance metrics, and in particular the measure B_W based on the weak notion of balance performs significantly better than any of the strong balance measures.

One must be a little careful about these results, however, because, as mentioned previously, positive edges outnumber negative ones by a wide margin in most cases. This means that one can achieve quite high prediction accuracy simply by guessing that every edge is positive. The magenta curve in Fig. 2.3 represents this baseline level of accuracy and it is against this curve that the others should be judged. Thus, for example, the measure of EB, which gave generally good performance in Fig. 2.1, performs least well in terms of sign prediction accuracy and in some cases is actually below the baseline estimate, particularly in the latter half of the data set. Meanwhile, the weak balance measure B_W substantially outperforms the other measures and the baseline, and appears to give the best sign prediction performance of the measures

considered.

Figure 2.4 shows an alternative measure of prediction performance, the normalized mutual information [124]. Often used to quantify the success of community detection algorithms on networks, normalized mutual information (NMI) is an information theoretic measure that reflects the amount of information about the true signs of edges that is contained in the predicted signs. If the predicted signs match the true signs exactly, the NMI is 1; if there is no correlation between true and predicted signs it is zero.

The (unnormalized) mutual information between true signs s_t and predicted signs s_p is defined as

$$I(s_t; s_p) = \sum_{\substack{s_t=\pm 1 \\ s_p=\pm 1}} P(s_t, s_p) \log \frac{P(s_t, s_p)}{P(s_t)P(s_p)}. \tag{2.19}$$

The joint probabilities $P(s_t = \pm 1, s_p = \pm 1)$ can be calculated straightforwardly by simply counting the fraction of times in our tests that each of the four possible configurations of the true and predicted signs occurs, and similarly for the marginal probabilities $P(s_t = \pm 1)$ and $P(s_p = \pm 1)$. The *normalized* mutual information is then calculated by dividing the unnormalized value by the average of the entropy $H(s) = -\sum_s P(s) \log P(s)$ of the two variables s_t and s_p [124]:

$$\text{NMI} = \frac{I(s_t; s_p)}{\frac{1}{2}[H(s_t) + H(s_p)]}. \tag{2.20}$$

This ensures that the normalized value falls between zero and one.

As shown in Fig. 2.4, the normalized mutual information for sign prediction using all four of our balance measures is better than the baseline estimate made by simply guessing that all edges have the majority positive sign—the latter automatically gets

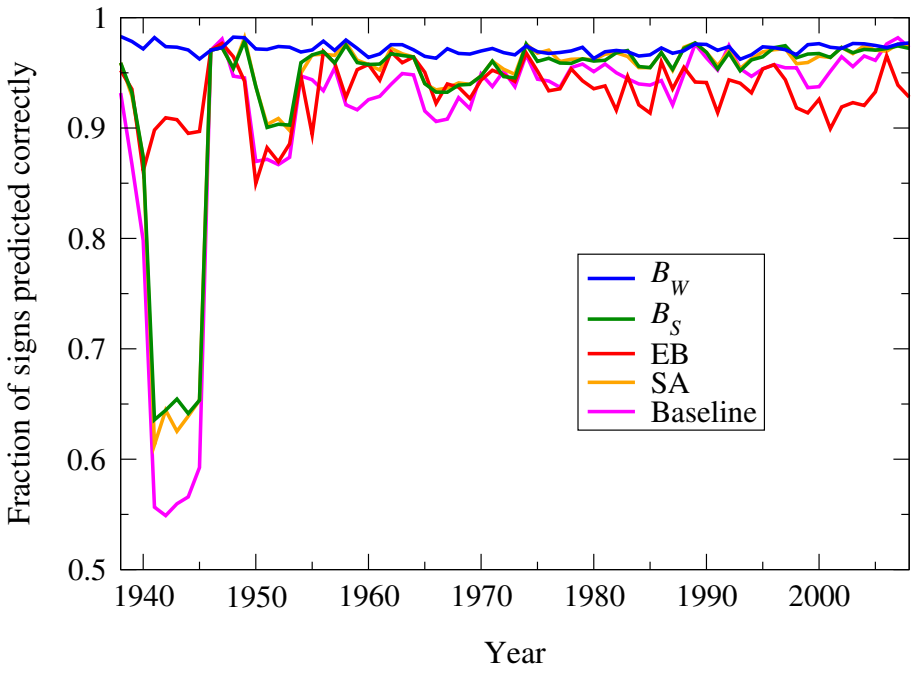

Fig. 2.3. Fraction of signs predicted correctly for each of the international relations networks in the single sign prediction task, using each of the four balance measures studied here.

an NMI of zero, since it is completely uncorrelated with the true signs of the edges. Again the weak balance measure B_W does best in most years, in some cases by a wide margin.

Comparing our results from this section with those on overall balance from Section 2.3.1, we see something of a mixed picture. Overall balance appears to be similar for all metrics, except during the Second World War era, when there were two primary factions and the strong notion of balance seems to be favored. Our sign prediction results, on the other hand, seem to give a clear edge to the weak notion of balance, even during the war years. What we can say with some clarity, however, is that these networks are more balanced than one would expect on the basis of chance, and one can use this fact to predict the signs of edges with good accuracy.

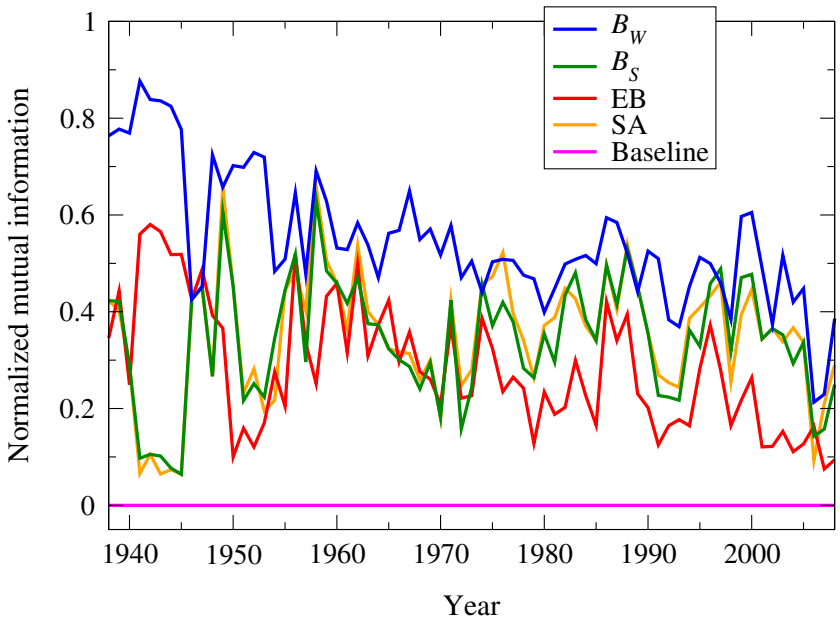

Fig. 2.4. Success in the single sign prediction task, as measured using normalized mutual information, for each of the four balance measures studied here.

2.3.3. Prediction of multiple edge signs

In the calculations of the previous section, we tested our ability to predict a single unknown sign in an otherwise known network. This single sign prediction challenge has the advantage of being relatively computationally tractable, but, as we have argued, it is not entirely realistic. In real-world data sets it is likely that many signs will be missing from our network simultaneously, not just one, and hence we need a way to predict multiple signs simultaneously. We can approach the latter problem in a similar manner to single edge prediction, by selecting the combination of signs that gives the lowest imbalance, but the calculation rapidly becomes intractable as the number k of signs to be predicted becomes large, since there are 2^k different combinations of signs to test.

To get around this issue, we employ simulated annealing to optimize balance over

sign configurations. We perform a Markov chain Monte Carlo simulation in which we initially give random values to all of the unknown signs, then we repeatedly select one of them at random and consider flipping its value, from positive to negative or vice versa. We can use any one of our imbalance metrics as an energy function and accept or reject sign flips using a standard Metropolis–Hastings acceptance probability with temperature T. We then lower the temperature from a high initial value T_0 according to the exponential cooling schedule $T = T_0 \, \mathrm{e}^{-t/\tau}$, where t is the number of Monte Carlo steps performed so far and τ is the annealing time-scale. The calculation ends when the state of the system stops changing and we take the final state to be our prediction of the unknown signs.

For the calculations presented here we use parameter values $T_0 = 0.1$ and $\tau = 10^4$ and run our calculations for 10^6 Monte Carlo steps. For each network studied, we remove varying fractions of the signs and then attempt to predict those removed, repeating the entire calculation 100 times for each fraction. For the imbalance measures used in this study the calculation can be sped up significantly by rapidly computing the new energy value upon the flip of a sign using the Woodbury or matrix determinant formulas again. Here we focus specifically on the measures B_W and B_S. Since these measures are constructed in an identical manner apart from the criteria they use for balanced loops, they give us an opportunity to perform an apples-to-apples comparison of strong and weak notions of balance, to see which gives better sign prediction. Similar calculations would, however, certainly be possible for the EB and SA metrics considered in previous sections.

Figures 2.5, 2.6, 2.7, and 2.8 show accuracy and NMI results from calculations for three of our international relations networks, corresponding to the years 1944 (during the Second World War, where 43% of signs are negative), 1950 (a few years afterward,

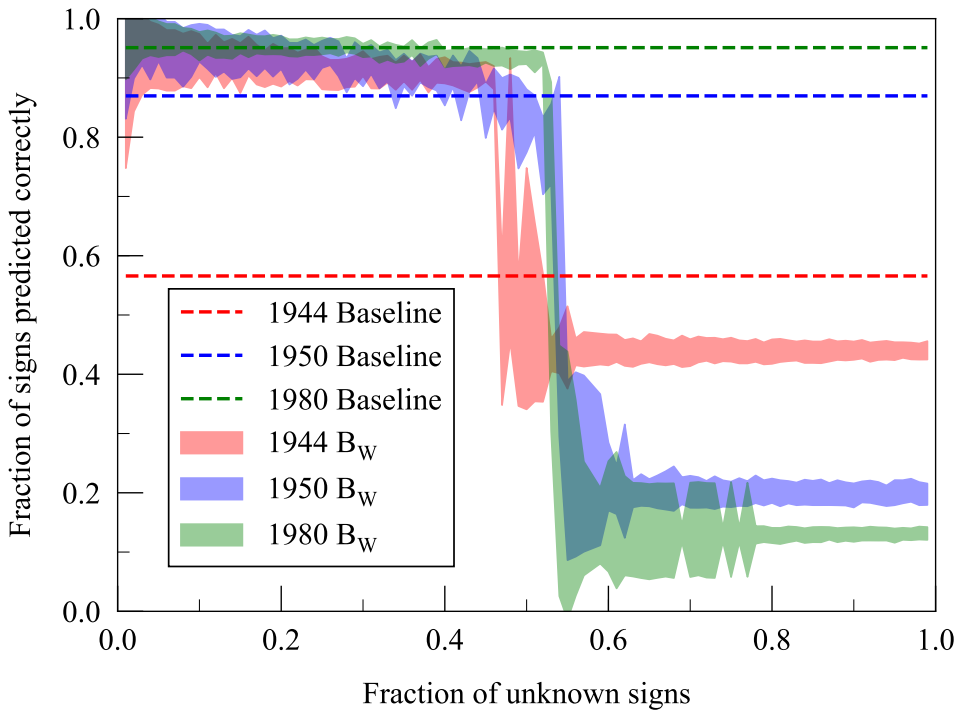

Fig. 2.5. Fraction of signs predicted correctly in the multiple sign prediction task using the weak balance measure B_W, as a function of the fraction of unknown signs for the international relations networks in the years 1944, 1950, and 1980, along with baseline levels derived by simply assuming all signs to be positive. Bands indicate 1σ errors calculated from the distribution of values over 100 randomized repetitions of the calculation.

where 13% of signs are negative), and 1980 (relative peace, where 5% of signs are negative). Each plot shows three separate curves for the three networks, as a function of the fraction of signs removed from the network. For the accuracy plots we also show the baselines set by assuming that all unknown signs are positive. (For the NMI plots the equivalent baselines are by definition at zero.)

As the fraction of signs removed gets larger (and hence the amount of information remaining to learn from gets smaller) we naturally expect the performance of the algorithm to fall off. Figures 2.5 and 2.6 show results for the weak balance measure B_W

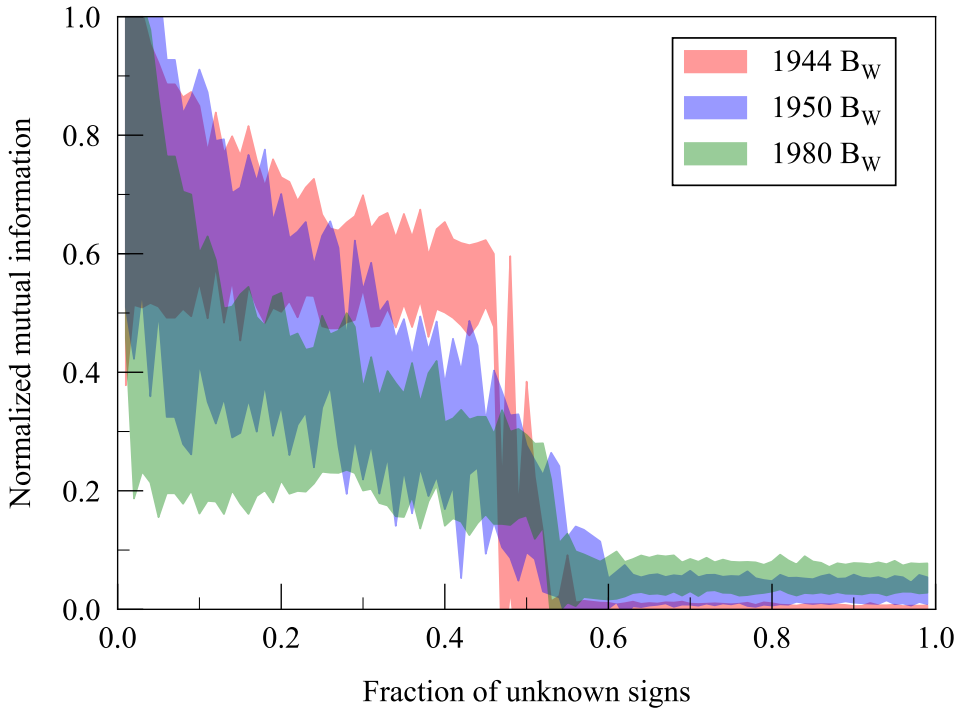

Fig. 2.6. Normalized mutual information for the multiple sign prediction task using the weak balance measure B_W, as a function of the fraction of unknown signs for the international relations networks in the years 1944, 1950, and 1980. The baseline level of normalized mutual information if we guess all signs to be positive is zero by definition. Bands indicate 1σ errors calculated from the distribution of NMI values over 100 randomized repetitions of the calculation.

and reveal that predictions are reasonably accurate for all three years studied for fractions of predicted signs up to about 50%, although the baseline accuracy for 1980 is so high that it is comparable with the predictions. (This is simply because a very large fraction of signs are positive in this network.) Normalized mutual information is also well above the baseline level of zero for fractions of predicted signs up to about 50%. Beyond the 50% mark, however, prediction accuracy rapidly falls to close to zero.

Figures 2.7 and 2.8 show the corresponding results for the strong balance measure B_S, and comparing the results for the two measures reveals an interesting overall

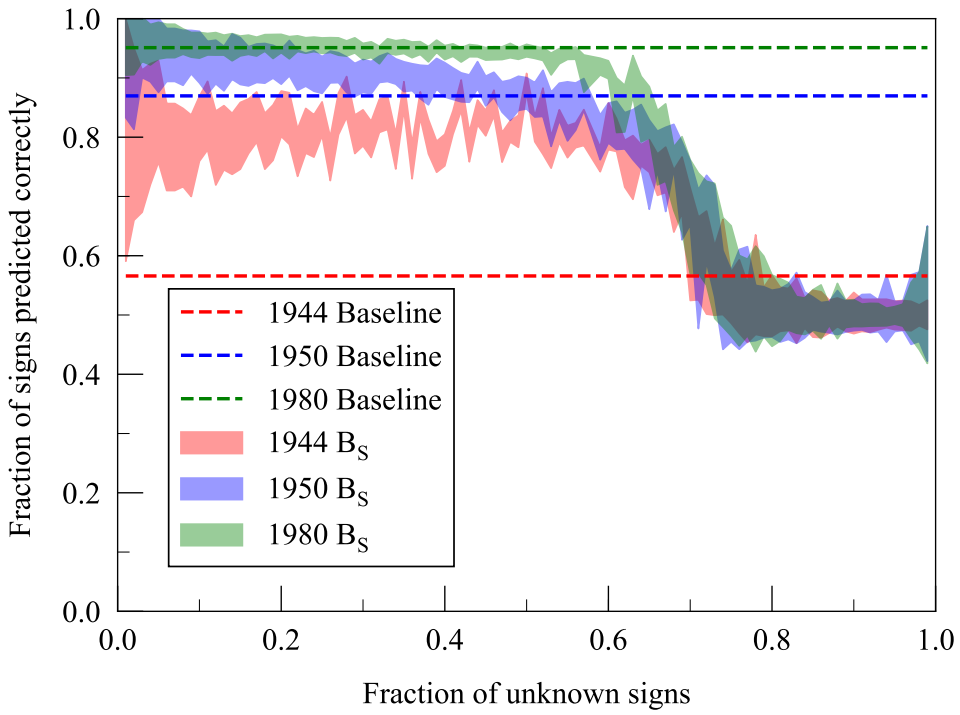

Fig. 2.7. Fraction of signs predicted correctly in the multiple sign prediction task using the strong balance measure B_S, as a function of the fraction of unknown signs for the international relations networks in the years 1944, 1950, and 1980.

picture. The weak measure does better when predicting smaller numbers of signs, but suddenly fails around the 50% mark, beyond which it does no better (in fact worse) than chance. The strong measure, by contrast, does less well when fewer than 50% of signs are removed, but manages at least modestly good performance well beyond the 50% point, thereby outperforming the weak measure in this regime (although it is still not very good). These trends are especially clear in the 1944 network, for which arguably the strong measure makes more sense since, as discussed earlier, international relations were dominated by two main factions during that era.

The failure of the weak balance metric to predict unknown signs beyond about the 50% mark is particularly interesting. It arises through a competition between two

different minima of the metric. One minimum approximately corresponds to the true assignment of signs, and if the algorithm finds this minimum it will succeed, at least partially, in the sign prediction task. The other minimum is a trivial one in which all, or almost all, unknown signs are negative. If the fraction of unknown signs is large enough, the latter state will contribute at least two negative signs to most closed loops in the network, meaning that most loops are balanced (according to the weak definition) and hence our imbalance score will approach its lowest possible value of zero. As the fraction of unknown signs grows, there comes a point at which this trivial minimum outcompetes the nontrivial one and the algorithm no longer predicts signs with success any better than chance. This point—the discontinuity we see in Fig. 2.6—is in effect a zero-temperature first-order phase transition between competing ground states. No similar argument applies to the strong balance measure, and hence we see no sharp phase transition in that case.

Overall, we conclude that successful prediction of multiple edge signs is possible using our balance measures, with the weak notion of balance again giving better performance than the strong notion, at least up to the phase transition mentioned above, beyond which the strong balance measure is a better choice. In the particular networks examined here, performance is stronger for the years 1944 and 1950 than for 1980, perhaps because of the starker conflicts and alliances during and immediately after the war, compared with the relative peace of the early 1980s.

2.4. Conclusion

In this chapter, we have studied the phenomenon of structural balance in signed networks, whereby some configurations of signed edges are more common than oth-

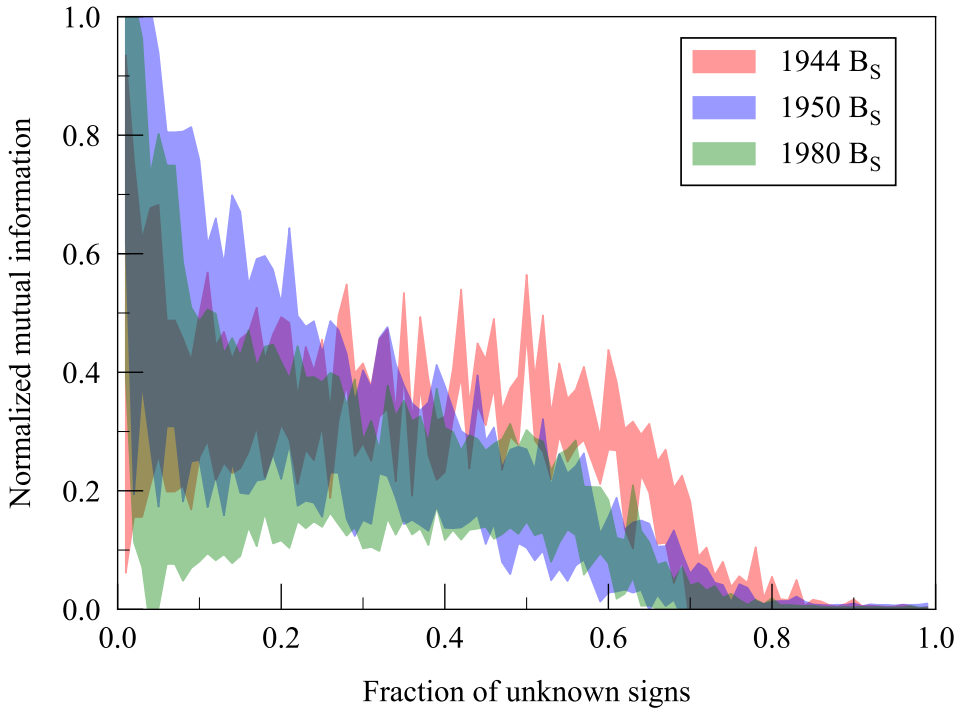

Fig. 2.8. Normalized mutual information for the multiple sign prediction task using the strong balance measure B_S, as a function of the fraction of unknown signs for the international relations networks in the years 1944, 1950, and 1980.

ers. We have proposed two measures of structural balance based on previously hypothesized notions of "weak" and "strong" balance and compared their performance against each other and previously proposed measures in a number of tasks. Specifically, we have examined the behavior of the various measures on two distinct sets of networks representing alliances and conflicts between countries during the 20th and 21st centuries, as well as university freshman cohort relationships, testing in the first instance to see simply by which measures these networks are most balanced. We find that all measures show a significant level of balance in all of the networks we study.

We further test our measures on the international relations data by comparing their ability to predict unknown edge signs in a set of cross-validation experiments, in

which we remove either a single sign or multiple signs from the network and attempt to predict the missing sign(s) by choosing those values that maximize balance by each of our metrics. We find that prediction of unknown signs is possible, with accuracy substantially better than a random guess, and in particular that our measure based on the weak notion of balance performs well in practice.

Chapter 3.

Multiscalar Diversity in Networks with Distributional Metadata

In this chapter, we return to discussing methods for characterizing structure in networks with metadata, this time looking at spatial networks with distribution-valued metadata. Perhaps the most natural setting in which this type of data arises is when analyzing socioeconomic data such as distributions of income or ethnicity across space, and so we focus our attention on this application. We will see that approaching spatial socioeconomic data from a network lens will reveal new patterns that are masked by variability in spatial scales, and that this perspective can also provide results that complement findings from more traditional spatial analysis.

3.1. Introduction

From analyzing demographic polling behavior [125], to epidemic vulnerability of populations [126], to disparities in access to nutritious food [127], spatial data on social

and economic attributes of populations is central to many problems in modern data-driven science. In particular, assessing the extent to which socioeconomic properties differ across regions of space is an important topic for understanding the spatial dynamics of production and consumption [128], the manifestation of segregation [129], and the spatial decomposition of inequality [130]. There has thus been extensive research to understand how socioeconomic indicators fluctuate across space, which has involved the development of many sophisticated mathematical techniques to quantify variation in spatial data.

A major challenge with these methods is determining the scale at which to probe spatial variations, noting that populations tend to disperse heterogeneously across space [131]. Extreme spatial inhomogeneity in population density causes inconsistencies in interpretability of results based solely on distance, as the pace of economic activity is largely determined by population [132], and space is primarily relevant insofar as it relates to the number of "intervening opportunities" it provides economic agents with [133]. As a result, various methods have been designed to account for heterogeneous populations in the analysis of spatial data, some of which include density-equalizing maps [134] and methods based on dasymetric mapping [135]. As an additional complication, there is no apriori way to aggregate regions of space for statistical analysis (an issue that is more precisely quantified by the Modifiable Areal Unit Problem, or MAUP for short) [136]. Consequently, finding suitable scales for various problems in spatial analysis is an open problem that has received extensive interest due the sensitivity of results to the chosen scale [137]. To make matters worse, policy interventions take place at the level of artificial government-designated boundaries, and so analysis that ignores these boundaries may be irrel-

evant for certain implementation-driven studies [138]. Here, we assess relationships between official boundaries (census tracts) in a network-based manner, circumventing the aforementioned spatial issues by considering topological distance rather than geographic distance, which also allows for the development of insights at the scale of regions designated for policy intervention.

Another important avenue of research investigates what measures to use to quantify spatial disparities in population data. For nominal distributions, such as race or religious affiliation, there are a wide variety of measures of qualitative variation that take the form of segregation indices between disjoint groups [139]. Some of these indices can also be used with ordinal or interval data such as population counts over income brackets or education levels [129], but few of these measures have the flexibility to accommodate all types of data on the same scale or generalize to more than two regions. Measures based on information theory can also be effectively applied to distributional socioeconomic data [140, 141], having the additional benefit of being founded in fundamental statistical principles, and allowing in some cases an extension to multiple distributions. We develop here a novel approach based on the Generalized Jensen-Shannon Divergence (GJSD) [142] to compare distributional data, which has a number of advantages over other approaches, including flexibility for all distributional data types and an intuitive theoretical interpretation.

We note other approaches proposed to analyze spatial data using networks or information theoretic principles, as there has been similar research in regional science, economic and political geography, urban planning, and spatial analysis. There has been extensive work on using spatial network methods in urban science [143, 144],

with a focus oriented mostly towards the structure of urban form and the dynamics of urban growth. Numerous methods based on spatial aggregation of neighboring regions within the context of multiscalar diversity indices have been developed to assess the spatial manifestation of diversity [145, 146], but rarely do these accommodate distributions or multiple data types. Additionally, there is a body of research constructing spatial correlation and aggregation methods based on information theoretic measures [147, 148]. However, these analyses thus far have been limited primarily to racial segregation and ecological diversity, and focus on relationships between individual regional entities within larger clusters rather than multi-distribution correlations as in this study. Furthermore, these measures may not be as easily interpreted in terms of a simple statistical process, in contrast to our method, and many of these measures are not adaptable to all data types, which limits their capability in comparative analysis.

In this chapter, we develop a novel approach for studying spatial variation in distributional socioeconomic data based on regional adjacency networks and information theory, and apply our methodology to the network of adjacent census tracts in the continental US through a few experiments. We first examine two-point correlations in our distributional distance measure with respect to path length across the adjacency network, finding a universal decay pattern with similar scaling exponents and finite size cutoffs across a variety of socioeconomic attributes. We also use this methodology to assess disparity with respect to various socioeconomic attributes across US counties by generalizing our measure to the comparison of more than two regions, finding high regional dependence and correlations in our measure for multiple variables. Finally, we discuss a new means for spatial aggregation of regions through

community detection at multiple resolutions based on our measure, clustering the census tract network for the city of Chicago into meaningful regions of homogeneous socioeconomic characteristics at different cluster size scales. Our methods provide a new means for analyzing spatial variation in all types of distributional data within a universal framework that circumvents limitations in traditional spatial measures. The results from our experiments point to new ways of thinking about how socioeconomic characteristics manifest across space, and can be applied to a wide range of problems across the social sciences.

3.2. Methods

3.2.1. Census tract data and network construction

In order to study a wide variety of socioeconomic attributes at high spatial resolution, we use US Census data at the tract level from the American Community Survey in 2018 [149, 150]. The American Community Survey continuously samples US households to collect data on various socioeconomic and demographic characteristics of the population, and it is the largest survey at the household level that is conducted by the Census Bureau. We choose to analyze data at the level of census tracts because they encapsulate highly localized populations, represent officially designated regions relevant for policy intervention [151, 152], and have roughly equal populations (the 25th and 75th percentiles in terms of population are 2971 and 5572 for the set of tracts used in the analysis). We aggregate distributional data on educational attainment, home price, income, industry of occupation, and race in order to assess spatial variability across a range of different variables. The techniques we develop can be adapted for

continuous distributional data, but here we use the available binned data for housing prices and incomes, leaving to future work the estimation of the full corresponding continuous distributions, as this is a difficult problem on its own [153, 154].

In order to quantify variation in the discrete distribution of a variable X across tracts, we encode its possible values as a vector q_X, which may be nominal, ordinal, or interval in nature depending on the variable X being analyzed. For census tract i, we denote its distributional vector of values for the variable X as $q_X^{(i)}$, with the particular value for an entry x denoted $q_X^{(i)}(x)$. These tract distributional vectors are normalized, and satisfy $\sum_x q_X^{(i)}(x) = 1$ for all tracts i and variables X, making $q_X^{(i)}(x)$ a probability mass function over realizations x of X. For example, if in census tract 5 there are 300 persons classified as *Asian* out of 1000 total persons, then $q_{race}^{(5)}(Asian) = 0.30$. Details on the variables analyzed are given in Table 3.1.

The nearest-neighbor network representation for census tracts is constructed with TIGER shapefile data [155], and two tracts are neighbors in the network if they share a common length of border or a corner. Only tracts in the continental US were considered for this analysis in order to ensure a single connected component for two-point correlation analyses. After removing tracts with corrupted or incomplete data, the final network had $70,201$ nodes and $197,841$ edges (for an average degree of 5.6). The overall goal in terms of practical relevance of the proposed methods is for local spatially targeted interventions (e.g. at the scale of counties, cities, or neighborhoods, with tracts as the fundamental subdivision), and so we are only presently interested in relatively short range correlations, hence the choice to construct the underlying network based on spatial adjacency. However, the method we present for comparing

local distributions of socioeconomic variables can be applied to any pair of regions (whether or not they are adjacent), which is in fact what is done for our two-point correlation analysis, and so any network structure signifying a relationship between two regions could be used in this framework. For instance, one could construct the network based on population migration flows from region to region, which would no longer necessarily have geographically localized edges, but could be used to see whether or not people move homes between regions with similar or different socioeconomic properties.

As the analyses performed in this study are intentionally topological in nature, rather than geographic, we do not focus on spatial dimensions. However, for better contextualization of our results for those unfamiliar with the spatial extent of subdivisions within the US, we report summary statistics from our network dataset here. For the subset of tracts used in the analysis, the distribution of land areas is heavily right-skewed, with the tracts in the 10th percentile, median, and 90th percentile having areas of 1.0 km^2, 6.4 km^2, and 269.2 km^2 respectively. If we consider the set of tracts kept in the filtered dataset, and construct their (potentially incomplete) associated counties, the distribution of land areas is also right-skewed, with the counties in the 10th percentile, median, and 90th percentile having areas of 953.0 km^2, 1911.4 km^2, and 4863.0 km^2 respectively. The high level of heterogeneity we see in the land area statistics at both the tract and county level further illustrates the utility of an approach to socioeconomic correlations that is spatial scale-independent, as adminstratively equivalent regions clearly can have drastically different sizes.

3.2.2. Generalized Jensen-Shannon divergence

Due to its desirable properties as a distributional distance measure, which we discuss in more detail, the Generalized Jensen-Shannon Divergence (GJSD) has gained popularity for applications across disciplines, from quantum physics [156], to genomics [157], and even to history [158]. For our purposes, the GJSD will allow us to distinguish distributional data across census tracts in a meaningful way, which can be understood in terms of the following process.

Suppose we have two spatial regions, region 1 and region 2, and we want to determine how similar these regions are with respect to a socioeconomic variable X. We assume that their respective populations n_1 and n_2 are known, as well as the distributions $q_X^{(1)}(x)$ and $q_X^{(2)}(x)$ defined in Sec. 3.1. One way to think about how the populations in regions 1 and 2 differ in their composition of the attribute X is to consider the situation where there was no artificial line drawn between regions 1 and 2, and instead we had just decided to consider them one single "super-region". We can then ask the question: How different is the distribution of X across the population in this super-region than in its individual sub-regions? Rather than naively comparing the distributions $q^{(1)}$ and $q^{(2)}$ directly, this perspective accounts for the population difference between the regions, and will also allow us to address in a natural way the increase in regional homogeneity we get by separating these regions.

From an information theoretic perspective, we can quantify the homogeneity of attribute X within a population by its *average information content* (or *surprisal*), in other words how unpredictable it is. For instance, if a population has relatively equal frac-

tions of people from each race, it is difficult to guess what any given person's race is, and the amount of "information" we gain by finding out each person's race is relatively high on average. However, if nearly everyone is of a single race, it is very easy to guess an individual's race, and we are on average very "unsurprised" upon each discovery of the race of a randomly selected individual in this population. For our thought experiment, we can determine the homogeneity gain we achieve by separating regions 1 and 2 by computing how much the average information content of attribute X in the population is reduced after the split of the super-region.

The average information content of a random variable with probability distribution $q(x)$ is given by its entropy, $H[q(x)]$, where H is the Shannon entropy functional

$$H[q(x)] = -\sum_x q(x) \log q(x) \tag{3.1}$$

and $\log q(x)$ is the information content of an observation of x [159]. Thus, the average information content of attribute X in the super-region population is given by

$$H\left[Q_X^{(12)}(x)\right] = -\sum_x Q_X^{(12)}(x) \log\left(Q_X^{(12)}(x)\right), \tag{3.2}$$

where

$$Q_X^{(12)}(x) = \frac{n_1}{n_1+n_2} q_X^{(1)}(x) + \frac{n_2}{n_1+n_2} q_X^{(2)}(x) \tag{3.3}$$

is the empirical probability mass function of X in the super-region. Now, if regions 1 and 2 are split, then we can associate to any individual in the super-region a label $i = 1, 2$ denoting the region they are from, which will necessarily reduce our uncer-

tainty about their value of X on average. Then, in a random experiment to survey the same population about X, we would know the region k that each person we sample is from, and thus the information content associated with each observation we make is $\log q_X^{(k)}(x)$ rather than $\log Q_X^{(12)}(x)$. The average information content $H'\left[Q_X^{(12)}(x)\right]$ of X after the regional split is then given by the weighted average

$$H'\left[Q_X^{(12)}(x)\right] = \frac{n_1}{n_1+n_2} H[q_X^{(1)}(x)] + \frac{n_2}{n_1+n_2} H[q_X^{(2)}(x)], \tag{3.4}$$

and the reduction in average information content from splitting the regions is given by the difference

$$J_X^{(12)} = H\left[Q_X^{(12)}(x)\right] - H'\left[Q_X^{(12)}(x)\right]. \tag{3.5}$$

Generalizing our argument to the merging of $m \geq 2$ regions, we have that the reduction in average information content by the separation of m adjacent regions is given by

$$J_X^{(1,\ldots,m)} = H\left[Q_X^{(1,\ldots,m)}(x)\right] - \sum_{k=1}^{m} \pi_k H[q_X^{(k)}(x)], \tag{3.6}$$

where

$$\pi_k = \frac{n_k}{\sum_{k'=1}^{m} n_{k'}} \tag{3.7}$$

(with n_k the population of region k) and

$$Q_X^{(1,\ldots,m)}(x) = \sum_{k=1}^{m} \pi_k q_X^{(k)}(x). \tag{3.8}$$

We can recognize now that Eq. 3.6 is equivalent to the *Generalized Jensen-Shannon Divergence* (GJSD), which is sometimes referred to as just the *Jensen-Shannon Divergence* for $m = 2$ [142].

Intuitively, if the distributions $\{q^{(k)}\}$ are all very similar, knowing which region that a person is from does not reduce our uncertainty about their value of X by much, and $J^{(1,..,m)}$ will be close to 0. On the other hand, if the $\{q^{(k)}\}$ are relatively different, then we can reduce our uncertainty about a person's value of X by a lot by knowing which region k they are from, and $J^{(1,...,m)}$ will be higher.

We know that Eq. 3.6 is bounded below by 0 due to the concavity of entropy, and this minimum is achieved when $q^{(k)} = q^{(k')}$ for all k, k', as merging the regions does not change our uncertainty about a person's value of X at all. On the other hand, the maximum value $J_{max}^{(1,..,m)}$ that Eq. 3.6 can take is

$$J_{max}^{(1,...,m)} = -\sum_{k=1}^{m} \pi_k \log \pi_k, \tag{3.9}$$

which happens when the $\{q^{(k)}\}$ are entirely non-overlapping in their regions of non-zero probability. We can see that this is the upper bound by rewriting Eq. 3.6 in a more illuminating manner as

$$J_X^{(1,...,m)} = \sum_{x,k} \pi_k q^{(k)}(x) \log \left[\frac{q^{(k)}(x)}{\sum_l \pi_l q^{(l)}(x)} \right], \tag{3.10}$$

and noting that $\log \left[\frac{q^{(k)}(x)}{\sum_l \pi_l q^{(l)}(x)} \right] \leq \log \left[\frac{1}{\pi_k} \right]$, with the equality holding when $q^{(l)}(x) = 0$ for all $l \neq k$, which is equivalent to the q's having disjoint nonzero support. Eq. 3.9 is

just the average uncertainty we have about which smaller region k a randomly chosen person from the super-region will come from.

We normalize Eq. 3.6 by the upper bound in Eq. 3.9 to enforce values to lie in $[0, 1]$, which allows us to compare tract similarities for regions with variable populations n_k. The final expression we use for distributional comparison across regions is then

$$L_X^{(1,\ldots,m)} = \frac{J_X^{(1,\ldots,m)}}{J_{max}^{(1,\ldots,m)}}. \tag{3.11}$$

This measure is easily adapted to any discrete variable X, which can be nominal, ordinal, or interval in nature, allowing for the application of Eq. 3.11 to a wide variety of problems. It can also be adapted to continuous distributions through approximations of the differential entropy. We note that for ordered data, Eq. 3.11 is only sensitive to how much the probability mass changes between distributions of interest, not to where it moves. In this sense, there are other appealing measures for comparing ordered data, such as variants of the earth-mover's distance [160]. However, Eq. 3.11 has a major advantage over such previous measures in that it can be used to compare results across all types distributional data on the same scale, and can also accommodate the inclusion of more than two distributions for comparison. In the following section, we perform multiple experiments on the tract adjacency network using Eq. 3.11, demonstrating new insights on spatial socioeconomic variability that can be gained through our methodology.

3.3. Results

3.3.1. Two-point correlations

Two-point correlation functions—a term used to refer generically to functions that measure some type of average correlation between points in a system as a function of the distance between them—are an invaluable tool for describing spatial data for systems as diverse as galaxy clusters [161], turbulent fluids [162], and earthquakes [163]. In more recent work, the concept of the two-point correlation function has been extended to networks [164–166], where it refers to computing correlations between the properties (in most cases, degree) of two nodes as a function of the shortest path distance between them.

Here, in order to assess the "scale" at which socioeconomic properties vary across the US, we compute a two-point correlation function for L_X (Eq. 3.11) between census tracts as a function of the number of network hops between them. The effective distance we are concerned with is then consistent with policy-relevant boundaries and roughly accounts for the heterogeneous population density across space (as tracts have relatively similar populations as discussed earlier). In other words, the total population of neighbors at path distance l or less from a focal tract is roughly the same for all tracts, as the degree distribution of the analyzed network is highly homogeneous as is characteristic of spatial networks in general.

In our case, the two-point correlation function $C_X(l)$ for socioeconomic attribute X

as a function of (unweighted) network geodesic distance l is given by

$$C_X(l) = \frac{1}{n(l)} \sum_{i<j} L_X^{(ij)} \delta_{d_{ij},l}, \tag{3.12}$$

where δ is the Kronecker delta function, $n(l)$ is the number of node pairs separated by shortest path distance l, and d_{ij} is the shortest path distance between tracts i and j in the adjacency network. $C_X(l)$ gives the average divergence $L_X^{(ij)}$ over all pairs of nodes (i, j) that are separated by l hops.

Calculating $C_X(l)$ exactly is difficult, as there are ~ 2.5 billion pairs of tracts $\{i, j\}$ in the network that contribute to the sum in Eq. 3.12 for a given X. We therefore opt for a sampling procedure to compute $C_X(l)$ approximately, traversing nodes j in the network up to a distance $l = 20$ starting at $1,000$ uniformly sampled focal tracts i, then computing the sum in Eq. 3.12 over sampled focal tracts i and traversed nodes j. A network distance of $l = 20$ corresponds to a spatial distance of 200 km, varying depending on the location of the central tract, and so captures spatial regions roughly of size $160,000$ km^2 (or about 2% of the land area of the continental US). Using this distance cutoff thus restricts our analysis to relatively concentrated regions, which may be more relevant for spatially targeted policy interventions.

In order to examine the scale over which correlations in each attribute decay, we analyze how quickly $C_X(l)$ approaches its asymptotic value $C_X(\infty)$ from its initial value $C_X(1)$ as we increase l. $C_X(\infty)$ is estimated as the average value of L_X over 10,000 tract pairs selected uniformly at random (which should draw primarily from node pairs at distances much greater than $l = 20$ based on the network structure).

Taking inspiration from the form of two-point correlations in spin systems, we can then fit the resulting data to the truncated power-law form

$$\tilde{C}_X(l) = l^{-\alpha} e^{-(l-1)/\beta}, \tag{3.13}$$

where

$$\tilde{C}_X(l) = \frac{C_X(\infty) - C_X(l)}{C_X(\infty) - C_X(1)}, \tag{3.14}$$

and we have rescaled $C_X \to \tilde{C}_X$ to account for the intercepts at $l = 1$ and $l = \infty$.

The scaling exponent α in Eq. 3.13 quantifies the rate of decay in correlation in the system as a function of distance (network hops), and the cutoff exponent β determines the distance scale (in terms of hops) over which correlation persists. A higher (more positive) value of α indicates a slower decay in correlations as we move away from a given tract, and a higher value of β indicates a longer distance over which tracts have correlated distributions with this reference tract. To extract the exponents α and β, the following ordinary least-squares fit is performed

$$\log \tilde{C}_X(l) = -\alpha \log l - \frac{(l-1)}{\beta} + \epsilon_l, \tag{3.15}$$

with ϵ_l a white noise process.

We plot the results of the fit in Fig. 3.1A, where we show the coefficient of determination r^2, the scaling exponent α, and the cutoff exponent β for the fit in Eq. 3.15 for each attribute. We can see that the curves for all attributes (apart from "industry",

which due to autocorrelated residuals has been suspected to follow a different decay form that we will not explore here) collapse quite well onto each other. This collapse is not only an indication of a good fit, but can possibly lead us to consider a more fundamental, attribute-independent mechanism behind the variation of different attributes X across regions, which we will discuss at the section's closing.

To investigate a potential consequence of the striking similarity in the decay of $\tilde{C}_X(l)$ across attributes X studied in Fig. 3.1A, we examine the correlations between the losses $L_X^{(ij)}$ and $L_{X'}^{(ij)}$ across edges (i,j) for all pairs of attributes (X, X'). Specifically, we analyze the monotonic dependence between losses using Spearman correlation, which relaxes the linearity assumption of Pearson correlation but also allows us to test for the significance of observed correlations [167]. Specifically, we compute

$$\rho\left(\{L_X^{(ij)} : (i,j) \in E\}, \{L_{X'}^{(ij)} : (i,j) \in E\}\right), \tag{3.16}$$

where E is the set of edges in the adjacency network, ρ is the Spearman correlation coefficient, and the arguments to ρ describe the vectors of measurements being correlated. We plot the results as a correlation matrix in Fig. 3.1B, where we can see relatively high correlations between most of the variables analyzed. The high correlations we see are consistent with associations seen in a multitude of previous economic and sociological studies [168–171], although our framework has the added benefit of using a single unified formalism to analyze all these associations. However, to get at underlying universal mechanisms behind observed socioeconomic data, we must go beyond solely demonstrating statistical associations between variables. The correlations seen in Fig. 3.1B may actually just be an artifact of a more fundamental process

determining the decays in Fig. 3.1A, and we can make some headway in uncovering this process (or processes) using reasoning inspired by urban scaling.

Traditional urban scaling posits that a wide variety of characteristics Y in a city can be predicted solely by the city's population P through relations of the form $Y \sim P^{\eta}$ for some exponent $\eta > 0$, which in practice holds up for a large number of cities and variables of interest [132]. The success of the urban scaling theory relies on a few key factors that are associated with a growing city population: denser organization of facilities and infrastructure, an accelerated pace of life, and increased interaction between agents and activities leading to specialization and innovation [172]. In practice, the data Y for some city-level characteristic (such as new patents or number of gas stations) is fit versus city population P for many different cities, yielding an estimate for the exponent η which we can interpret to gain an understanding of the fundamental processes contributing to the scaling behavior of Y. For instance, if $\eta > 1$ this says that Y grows superlinearly with P, which should be the case for quantities Y that show increasing returns with population (e.g. indicators of innovation such as new patents). On the other hand, $\eta < 1$ indicates economies of scale, or characteristics Y that decrease in unit cost as we increase the city's population (e.g. mobility-related infrastructure such as the number of gas stations). Perhaps the most important takeaway from traditional urban scaling analysis is that when we can collapse the behavior of a wide range of seemingly different socioeconomic systems into a single framework with few parameters, these parameters can help us understand basic universal processes underlying these superficially distinct variables.

We can use similar reasoning to interpret the results of Fig. 3.1A, except in this case

rather than the absolute quantity of a socioeconomic indicator we are analyzing correlations between distribution-valued quantities, and the fundamental covariate here is network distance l instead of population P. Based on their homogeneous populations and degrees, the total population in tracts at path distance l or less from a focal tract is very similar across tracts, and so l reflects the total population included as we encircle a focal tract at greater and greater radii. As space is a factor for socioeconomic processes mainly in that it provides a medium for interaction among people [173], this distance l may be a more fundamental quantity than standard spatial distance in how it determines socioeconomic activity, and so we may be able to explain the spatial variation in a wide variety of socioeconomic variables using simple functions of l such as Eq. 3.13. An alternative quantity to l could be derived from literally transforming space based on population to homogenize the population density, a concept which has inspired numerous interesting and informative mapping methods [134]. However, we are ultimately constrained by the basic spatial units designated for data aggregation (e.g. census tracts), and so here we treat these regions, hence l, as fundamental.

In the present case, we can see that the exponents β determining the network correlation cutoff scale are very similar for education, housing, income, and race, indicating that correlations in these regional distributions are non-negligible over a universal distance scale of ~ 30 hops. However, we see higher variation in the scaling exponents α, with race and housing decaying at a slower rate across the network than education and income. This suggests that perhaps the mechanisms that drive spatial differences in racial composition and local real estate values operate over larger distances than the mechanisms behind income or educational variability, at least in the

US.

The association between the spatial distributions of housing values and racial groups has been noted in numerous studies that address "redlining" and other processes that result in lower property values in neighborhoods with high minority populations [174]. The analyses in Fig. 3.1A may point to additional, more subtle mechanisms behind this inequality due to a significant difference in the scaling exponents for housing and race, as this observed discrepancy indicates that the scales over which housing and racial regional similarity decay are quite different. It is known that home values are also tied to local incomes, which in turn can result in high variability in housing prices due to the relative flexibility of wages and mobility of workers compared to supply-regulated housing [175]. Therefore, perhaps the interplay between the long-range correlated racial composition of the population and the comparatively short-range correlated income distributions plays a role in determining the moderate decay exponent α we see in the housing data. However, more definite conclusions and practical intervention strategies require a more contextualized analysis in conjunction with domain expertise.

3.3.2. County-level heterogeneity

To examine the regional diversity of a given socioeconomic variable, we employ Eq. 3.11, this time to all the tracts comprising each county within our dataset. More specifically, for each county we examine, we compute $L_X^{(t_1,t_2,\ldots)}$ with t_k the census tracts within the county subdivision and X the variable of interest. For notational convenience, we will use the notation $L_X^{(county)}$ from now on for this quantity. In or-

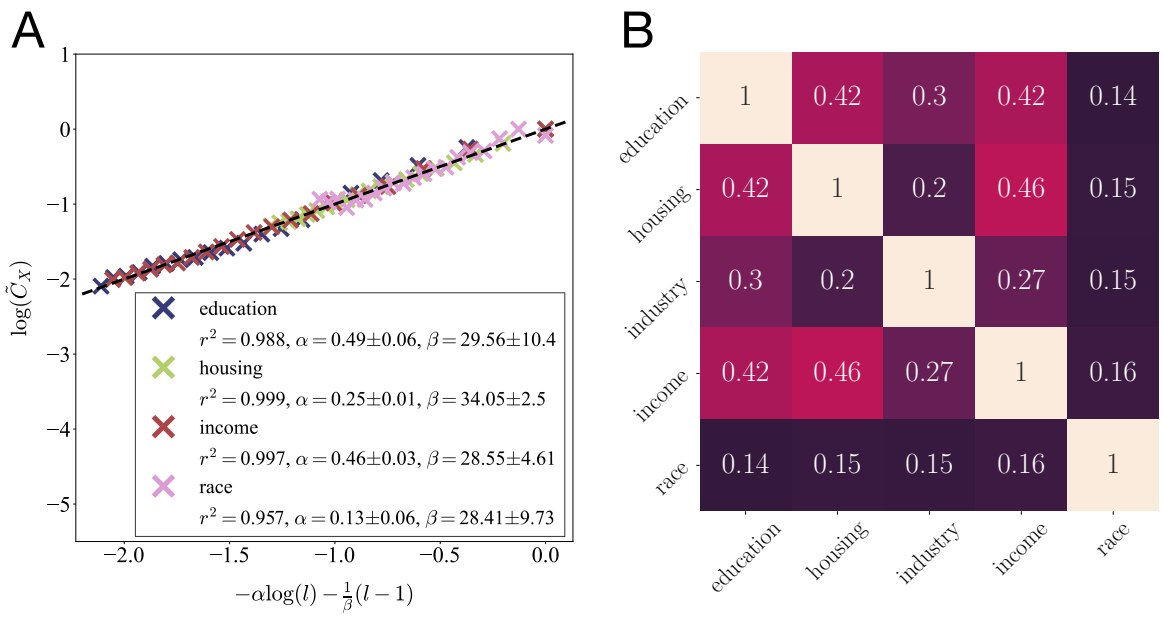

Fig. 3.1. Universal patterns in tract similarity across attributes. **(A)** Fit results for the two-point correlation functions $C_X(l)$ for attributes X in Table 3.1, with 95% confidence intervals around the scaling and cutoff exponents α and β. The line $y = x$ is plotted for reference, as a perfect scaling collapse maps all points onto this line. Eq. 3.15 is deemed a poor fit for $\tilde{C}_{industry}$ after a residual analysis, and so this result is omitted. **(B)** Spearman correlation matrix with respect to losses $L_X^{(ij)}$ across all edges (i, j), for all pairs of socioeconomic attributes used in our study. All correlations are highly statistically significant at the 1% significance level, with standard errors of ~ 0.001.

der to compare counties with varying numbers of constituent tracts on the same scale, we normalize for potential biases from the number of included tracts by using a bootstrapping procedure to compute z-scores for each county-level value $L_X^{(county)}$. To do this, for all county sizes (number of constituent tracts) K we compute the vectors μ_K and σ_K, which are the sample mean and standard deviation of $L_X^{(t_1,\ldots,t_K)}$ over 100 random samples of K tracts $t_1, \ldots, t_K$. Then, we calculate a standardized version of Eq.

3.11, $\tilde{L}$, for each county using

$$\tilde{L}_X^{(county)} = \frac{L_X^{(county)} - \mu_{|county|}}{\sigma_{|county|}}, \tag{3.17}$$

where $|county|$ is the number of tracts within the county. We will refer to Eq. 3.17 as a "disparity" measure, as higher values of $\tilde{L}_X^{(county)}$ indicate higher dissimilarity in a county's tract-level distributions of $\{q_X^{(t_k)}\}$ relative to what is expected in a randomized null model where the county's tracts are chosen at random. In practice, we will see that $\tilde{L}$ tends to be negative for most counties, and so in this case we should note that values of greater magnitude indicate higher *similarity* in the county-aggregated tracts than expected by chance.

As a first step in understanding county-level disparities across the US, we plot the distribution of $\tilde{L}_X^{(county)}$ over all counties for each socioeconomic attribute X in Fig. 3.2A as box-and-whisker plots. We can see that the distributions of all quantities tend strongly towards negative values, indicating that most counties have greater similarity in their tract-level distributions $\{q_X^{(t_k)}\}$ than expected in the null model. This is consistent with the spatial autocorrelation at short scales we see in socioeconomic variables in Fig. 3.1A, although these analyses in some sense provide a complimentary viewpoint. Here, rather than assessing the scales over which distributions of socioeconomic characteristics remain similar, as in Fig. 3.1, we are examining whether artificially drawn administrative boundaries are effective at capturing the homogeneity in these attributes. As counties have size scales much smaller than the area covered up to the typical correlation cutoff scale $l \sim 30$ from any reference tract, we expect that correlations between tract-level distributions will be relatively high within counties,

and so in this sense these results should be unsurprising.

Looking at Fig. 3.2A, we do see something perhaps unexpected though: there are lots of counties that are only slightly more (and sometimes less) homogeneous in their tract-level distributional data than we'd expect by chance. In particular, most of the values of $\tilde{L}_{race}^{(county)}$ are in the interval $[-2, 0]$, which means they are less than two standard deviations different in their disparity than expected with completely randomized tracts. This suggests that many counties in the US are relatively representative of the whole US in terms of racial composition, whereas there are relatively few counties with drastically different compositions. The same does not hold for housing though, for which around 75% of the counties studied had more than two standard deviations differentiating their disparity values from the null model expectation. This result indicates that there are relatively few counties with distributions of housing values that are diverse enough to reflect typical housing prices nationally.

To determine the association in the disparity values $\tilde{L}_X^{(county)}$ across counties, we plot the corresponding Spearman correlation matrix using the results from all counties studied. Similarly to Eq. 3.16, we compute

$$\rho\left(\{\tilde{L}_X^{(c_1)} : c_1 \in \text{counties}\}, \{\tilde{L}_X^{(c_2)} : c_2 \in \text{counties}\}\right). \tag{3.18}$$

The Spearman correlation matrix in Eq. 3.18 is shown in Fig. 3.2B, where we can see very high correlations between the within-county disparities, even higher than in the values of $L_X^{(ij)}$ shown in Fig. 3.1B. These correlations are similar in sign and relative magnitude (between attributes X) to those in Fig. 3.1B, but by aggregating tracts at

the county level rather than just assessing correlations over edges, we are effectively reducing noise by smoothing out local fluctuations, and so we see a major increase in the values of ρ. In other words, some individual edges (i, j) may have very different divergences $L_X^{(ij)}$ and $L_{X'}^{(ij)}$, but the effect of these outlier pairwise relationships is reduced when looking at distributions between tracts at the county-level. This noise reduction is only possible because, as discussed, the scale at which we are analyzing $\tilde{L}_X^{(county)}$ is smaller than the area associated with the correlation cutoff scales β found in Fig. 3.1A.

Finally, as a case study to visualize the geographic manifestation of these county-level disparities, we plot a heatmap of $\tilde{L}_{housing}^{(county)}$ across all counties studied in Fig. 3.2C. Here we can immediately see an interesting pattern: the county-level disparity in housing prices, when compared to the same number of randomly selected regions, is actually much *lower* along the coasts and metropolitan areas than it is elsewhere. Housing markets in coastal and metropolitan regions are typically seen as having high inequality due to the large variation in home and land values often seen in these areas [176, 177]. However, when assessed on a relative scale using distributions at the granularity of census tracts, we see a different story. In this case, we see that these coastal and metropolitan counties actually have quite similar distributions $q_{housing}^{(t)}$ across their constituent tracts t relative to more inland and rural counties. The primary reason for this may be that the heterogeneity in housing prices in dense, urban counties is primarily manifested at scales below our measurement precision: tracts themselves have house price distributions with high variance, but tracts in a given county tend to have relatively similar distributions. This is consistent with the low rate of spatial decay in housing correlations seen in Fig. 3.1A, as most tracts are urban

[178] and if most of the fluctuations persist at scales smaller than census tracts, we will see a relatively smooth correlation trend at larger scales. Due to the coarse binning of housing values, however, there is also a potential confounding factor here in that many of the home prices in expensive metropolitan and coastal regions fall into the highest bin in the corresponding census data ($> \$1,000,000$), and so variability due to fluctuations above this price threshold are suppressed when using census data to assess inequalities.

3.3.3. Regional clustering

As a final experiment using our measures, we detect communities at multiple size scales in the census tract subnetwork within the city of Chicago—a frequently used case study in socioeconomic diversity due to its rich history and abundance of available data [179, 180]—with the goal of constructing clusters that are relatively homogeneous with respect to each socioeconomic attribute. Optimal aggregation of spatial regions according to various criteria has been a longstanding problem of interest, and numerous approaches have been proposed to tackle this using networks with edges weighted by an attribute representing regional similarity. This approach has the added benefit that since community detection algorithms look for connected clusters of nodes, the clusters detected naturally tend to be contiguous, and thus relevant for spatially localized policy. Attributes used in previous studies include phone calls between regions [181], commuting flows [182], taxi trips [183], and similarity between individual within-region features like our own method [184].

In order to group the tract network into clusters that exhibit homogeneity with re-

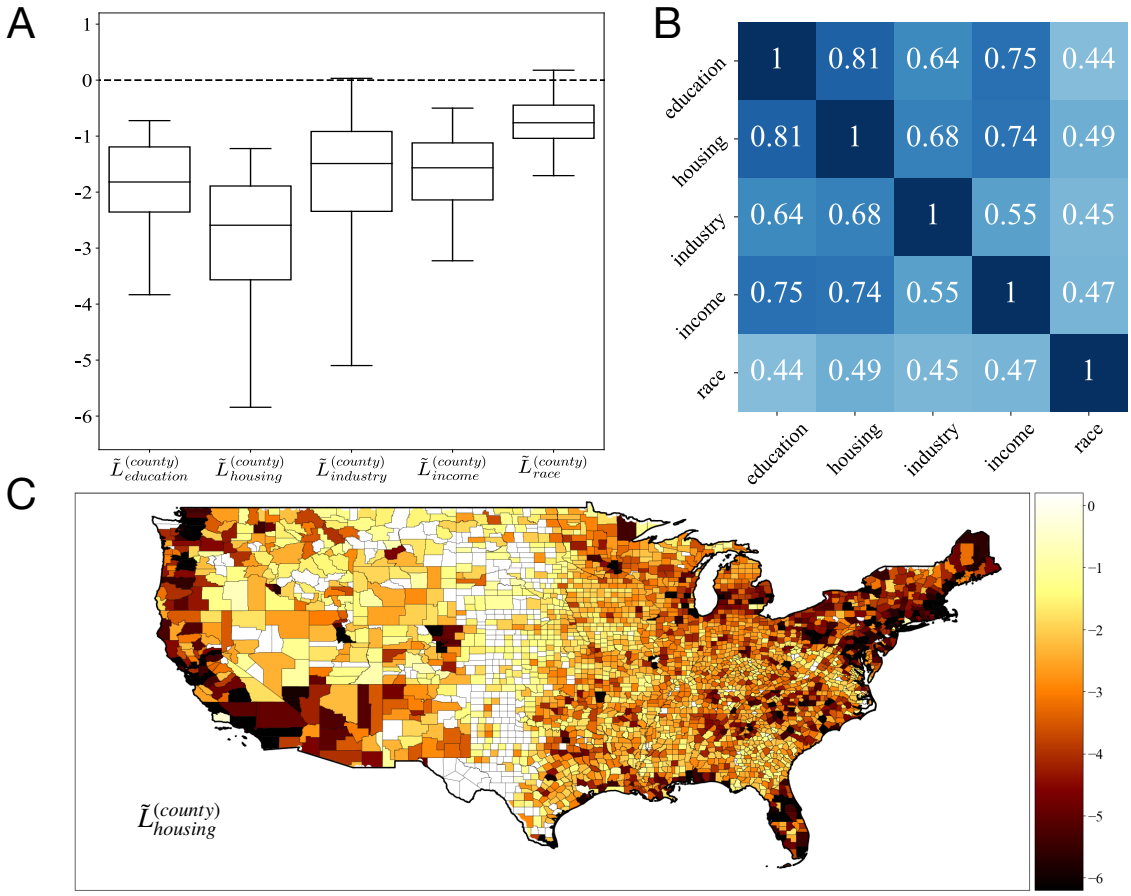

Fig. 3.2. County-level distributional disparity. **(A)** Distribution of $\tilde{L}_X$ for all measures X, showing trends towards negative values indicating lower within-county disparity than expected by chance in a randomized null model. Whiskers extend to the 5th and 95th percentiles. **(B)** Spearman correlation between county-level disparity measures $\tilde{L}_X^{(county)}$ for all pairs of attributes, showing a high positive association between all pairs of variables. All correlations are highly statistically significant at the 1% significance level, with standard errors of ~ 0.01. **(C)** Housing price disparities $\tilde{L}_{housing}^{(county)}$ for all counties in the continental United States. Values of $\tilde{L}$ indicate the degree to which counties' within-county distributional similarity in housing values differs from expectation (with more negative values associated with very high within-county similarity compared to expected).

spect to attribute X, we use $L_X^{(ij)}$ to construct edge weights w_{ij} for the network prior to performing community detection. However, we cannot not use $L_X^{(ij)}$ for edge weights directly, as community detection algorithms typically associate higher edge weight with higher node similarity, and $L_X^{(ij)}$ is constructed so that *lower* values indicate greater similarity across an edge (i, j). We thus employ a common transformation from the machine learning literature [185], which is to use an exponential kernel to map the values $L_X^{(ij)}$ to their associated edge weights w_{ij} in the network. The weight transformation can be written as

$$w_{ij} = e^{-\omega L_X^{(ij)}}, \tag{3.19}$$

where $\omega > 0$ is a tunable parameter that determines how much differentiation in the weights we will have across edges in the network. A value of $\omega \approx 0$ results in almost no differentiation between edge weights ($w_{ij} \approx 1$ for all edges), whereas $\omega >> 1$ results in an exaggerated difference in edge weights between edges with lower $L_X^{(ij)}$ and edges with higher $L_X^{(ij)}$. Any kernel mapping the unit interval to decreasing non-negative reals would suffice to construct the weights w_{ij}, but we opt for the exponential function here because it is particularly simple and only has one tunable parameter. For the experiments shown, we find a middle ground between the two extremes presented for ω, for each attribute-based clustering choosing a value of ω that results in a relatively uniform distribution of edge weights across $[0, 1]$. More specifically, for each attribute X we numerically approximate the ω that maximizes the entropy of the associated distribution of edge weights $e^{-\omega L_X^{(ij)}}$. A more principled method for choosing ω based on the application of interest is a subject is left to future work, but here we use this simple statistical procedure to avoid falling into one of the two cases presented,

where there is either no differentiation in the edge weights or only a handful of edges matter.

In order to detect communities in the Chicago subnetwork, we aim to find the community division $\mathbf{g} = \{g_i\}$ in the subnetwork such that the weighted modularity $Q_\gamma(\mathbf{g})$ is approximately optimized. The modularity $Q_\gamma(\mathbf{g})$ that we use here is defined by

$$Q_\gamma(\mathbf{g}) = \frac{1}{W} \sum_{ij} \left[\gamma w_{ij} - \frac{s_i s_j}{W} \right] \delta_{g_i, g_j}, \tag{3.20}$$

where W is the sum of edge weights in the network, $s_i = \sum_k w_{ik}$ is the sum of weights of edges attached to node i, and γ is a *resolution parameter* [186]. When $\gamma = 1$, Eq. 3.20 reduces to the standard modularity for weighted networks, but varying $\gamma \neq 1$ allows us to choose the importance given to w_{ij} relative to $\frac{s_i s_j}{W}$ (which is the approximate expected weight of w_{ij} through random rewiring). In particular, the larger we make γ, the more importance is given to the observed edge weights relative to the expected weights, and the community partitions $\mathbf{g}$ that maximize Eq. 3.20 will be larger. Thus, by varying ω we can tune how much influence differences in $L_X^{(ij)}$ across edges have, and by varying γ we can determine the characteristic cluster size. We use the Louvain Algorithm [187], a greedy optimization method, to find the partition $\mathbf{g}$ that approximately maximizes Eq. 3.20. There are numerous viable alternative methods but here we opt for the Louvain algorithm as it is fast and straightforward to implement. It is also important to note that we can perform regional aggregation with $L^{(ij)}$ in a manner where clusters are not likely to be contiguous, for instance by constructing a matrix from all pairwise values of $L^{(ij)}$ and performing one of various matrix-clustering techniques [188]. However, here we are interested in constructing contiguous clusters of

tracts in order to coarse grain the city into zones relevant for spatially targeted policy intervention, and so we use community detection to encourage contiguity of the clusters.

In Fig. 3.3 we show the results of our community detection analysis for the Chicago census tract subnetwork. In Fig. 3.3A-3.3C, we show the community divisions obtained for edge weights constructed using $L^{(ij)}_{income}$, at various resolutions γ. We can observe that increasing γ allows us to get a coarser view of the socioeconomic clusters present in the city, and can allow for delineation of super-regions at a desired scale. We also show the officially designated neighborhood boundaries (thick black lines) for Chicago (`https://data.cityofchicago.org/`) in order to visualize the consistencies and inconsistencies between our inferred communities and these officially delineated regions. We can see that in the intermediate regime $\gamma \sim 0.1$, some inferred communities are consistent with neighborhood boundaries, but others deviate significantly from these boundaries. This suggests that the officially designated neighborhood regions are somewhat consistent with homogeneous socioeconomic clusters, but there is room for improvement to these boundaries if the goal is to delineate socioeconomically homogeneous zones within the city (at least regarding income). Of course, there are numerous other factors, both socioeconomic and geographic, that would need to be accounted for in addition to the factors we analyze in order to draw effective policy-relevant boundaries in practice.

We also compute the Adjusted Mutual Information (AMI) between partitions obtained using different attributes X as well as the partition induced by the official neighborhood boundaries, in order to assess the consistency in the regions we ob-

tain when considering these different factors. As discussed in Sec. 2.3.2, the Mutual Information $MI(\mathbf{g}^{(1)}, \mathbf{g}^{(2)})$ is the amount of shared information (in an information theoretic sense) between the partitions $\mathbf{g}^{(1)}$ and $\mathbf{g}^{(2)}$, or more intuitively, the statistical uncertainty in each independent partition minus the statistical uncertainty when combined. More specifically, we have that

$$MI(\mathbf{g}^{(1)}, \mathbf{g}^{(2)}) = H[\mathbf{p}^{(1)}] + H[\mathbf{p}^{(2)}] - H[\mathbf{p}^{(12)}] = \sum_{s,t} p^{(12)}(s,t) \log \left[\frac{p^{(12)}(s,t)}{p^{(1)}(s) p^{(2)}(t)} \right], \tag{3.21}$$

where $p^{(1)}(s)$ is the fraction of nodes put into cluster s under configuration $\mathbf{g}^{(1)}$ (and similarly for $p^{(2)}(t)$), and $p^{(12)}(s,t)$ is the fraction of nodes simultaneously put into group s under configuration $\mathbf{g}^{(1)}$ and t under configuration $\mathbf{g}^{(2)}$. One drawback to using the mutual information, however, is that it gives systematically higher values as we increase the number of clusters, even for completely random cluster configurations [189]. One proposed correction to this is to use the AMI, given by

$$AMI(\mathbf{g}^{(1)}, \mathbf{g}^{(2)}) = \frac{MI(\mathbf{g}^{(1)}, \mathbf{g}^{(2)}) - \left\langle MI(\mathbf{g}^{(1)}, \mathbf{g}^{(2)}) \right\rangle}{\text{Max}(H[\mathbf{p}^{(1)}], H[\mathbf{p}^{(2)}]) - \left\langle MI(\mathbf{g}^{(1)}, \mathbf{g}^{(2)}) \right\rangle}, \tag{3.22}$$

where $\left\langle MI(\mathbf{g}^{(1)}, \mathbf{g}^{(2)}) \right\rangle$ is the expectation value of the mutual information in the null model where the number of nodes in each community is fixed and communities are generated randomly through permutations of labels. The AMI is equal to 0 if the partitions $\mathbf{g}^{(1)}$ and $\mathbf{g}^{(2)}$ share the amount of information we expect from random chance purely based on the sizes of their communities, and 1 if the partitions are identical.

In Fig. 3.3D, we plot the average AMI over all pairs of partitions using the five so-

cioeconomic attributes, as a function of the resolution parameter γ. We can see that there is a clear peak value of γ at which the five attributes share highly overlapping partitions. In practice, this could be used as a heuristic to tune γ for selecting the size scale of the clusters, if the goal is to select clusters that are highly homogeneous with respect to multiple socioeconomic attributes. It is interesting to note the clear scale sensitivity in this analysis: at certain scales, we can divide the city into zones that are relatively socioeconomically homogeneous in all variables studied, but at other scales, the city decomposes into regions with less overlap.

Fig. 3.3E shows the AMI matrix for the partitions obtained at $\gamma \approx 0.1$, the peak in Fig. 3.3D. We can see from this plot that all socioeconomic attributes are spatially clustered in quite similar patterns at this scale, and that all have high correlation with the official neighborhood boundaries as well. This is perhaps an endorsement for the neighborhood boundaries, as these results suggest that the scale at which the neighborhoods are drawn corresponds to the scale at which the socioeconomic clusters in the city are most similar. Taken together, the results from Fig. 3.3D and 3.3E may point to a new method for subdividing a city into different neighborhoods, which can be constructed easily based on any socioeconomic attribute and at any size scale.

3.4. Conclusion

In this chapter, we propose a new measure for analyzing socioeconomic data across spatial regions using concepts from network theory and information theory, which accommodates all forms of distributional data, has a natural extension to the com-

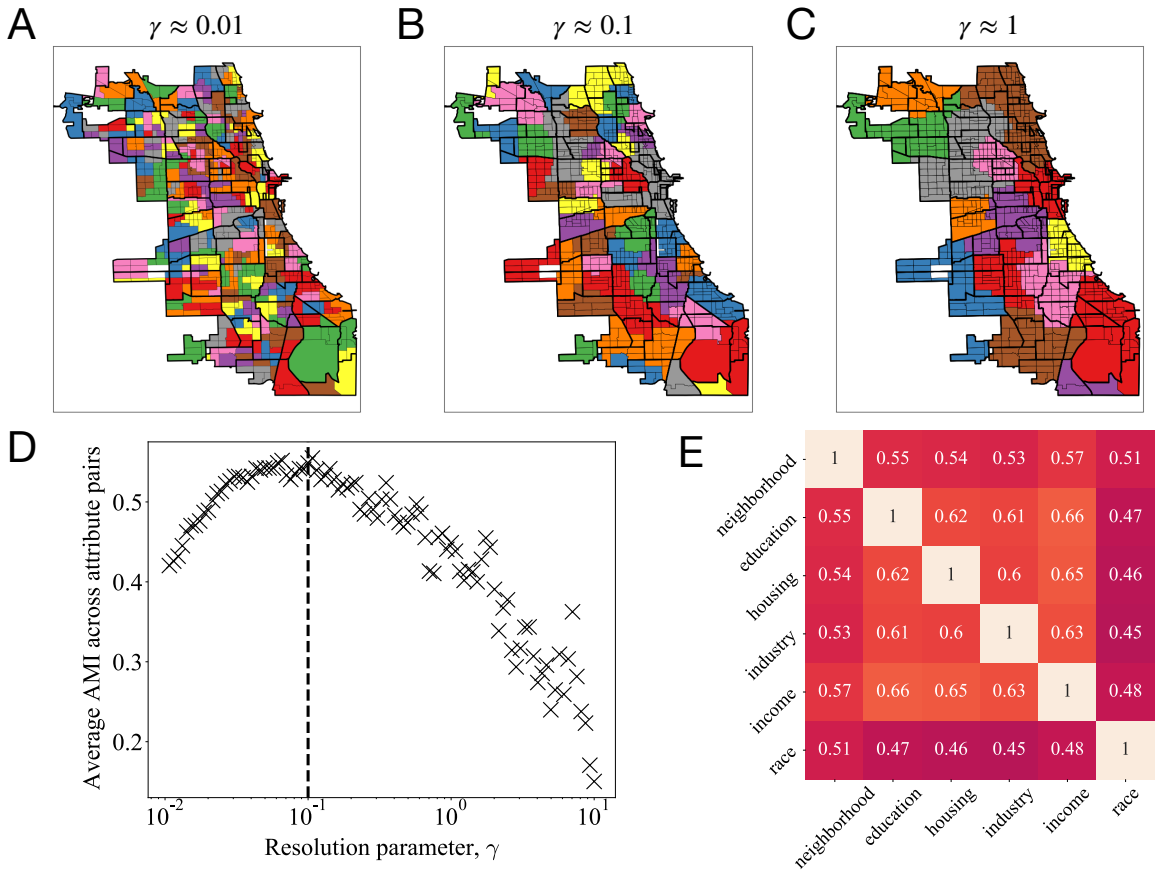

Fig. 3.3. Attribute-based regional clustering at multiple scales. **(A)-(C)** Clusters obtained through weighted community detection for census tracts (thin black lines) in Chicago with respect to income for various resolution parameters γ, displaying varying characteristic size and association with neighborhood boundaries (thick black lines). **(D)** Adjusted Mutual Information (AMI) between the partitions obtained by our community detection algorithm, as a function of γ and averaged over all pairwise combinations of the five studied socioeconomic variables. We see a clear peak value of $\gamma \approx 0.1$ at which the community divisions obtained through the five methods are highly correlated (dashed vertical line). **(E)** AMI matrix between partitions computed at this peak γ value, including the clusters obtained by grouping tracts by the neighborhood they most overlap with, indicating a high correlation between all of these partitions compared to what one expects by random chance based on their cluster sizes.

parison of more than two regions, and allows for policy-relevant analysis by considering officially delineated regions as fundamental spatial units. By analyzing spatial data from a topological lens, we can approach regional analysis issues from a relational perspective that avoids the longstanding issue of identifying appropriate spatial scales. We apply our framework in a series of experiments using the adjacency network of US census tracts to demonstrate the new insights we can gain with our methodology. We first find a universal decay pattern in various socioeconomic correlations as a function of path distance, as well as high statistical association between distributional similarities in adjacent tracts. We then aggregate tract-level distributions at the county level, finding again that distributional disparity measures are highly correlated, and also that there are relatively low levels of within-county inequality compared to what one would expect by aggregation of random tracts. Finally, we propose a clustering algorithm for regional aggregation into homogeneous socioeconomic clusters, finding that in practice the partitions obtained by our methodology have high overlap with accepted neighborhood delineations, as well as with each other across attributes. These applications illustrate the versatility of our methods, as well as the universality present in socioeconomic data when analyzed with a unified framework.

variable X	**support** $\{x\}$ **of** q_X	**ACS codes**
race	White Black or African American American Indian and Alaska Native Asian Native Hawaiian and Other Pacific Islander Other	DP05, 0059PE - 0064PE
income	Less than 10,000 10,000 - 15,000 15,000 - 25,000 25,000 - 35,000 35,000 - 50,000 50,000 - 75,000 75,000 - 100,000 100,000 - 150,000 150,000 - 200,000 Greater than 200,000	DP03, 0052PE - 0061PE
industry	Agriculture, forestry, fishing and hunting, and mining Construction Manufacturing Wholesale trade Retail trade Transportation and warehousing, and utilities Information Finance and insurance, and real estate and rental and leasing Professional, scientific, management, and administrative services Educational services, and health care and social assistance Arts, entertainment, recreation, accommodation, and food services Other services, except public administration Public administration	DP03, 0033PE - 0045PE
housing (value)	Less than 50,000 50,000 - 100,000 100,000 - 150,000 150,000 - 200,000 200,000 - 300,000 300,000 - 500,000 500,000 - 1,000,000 Greater than 1,000,000	DP04, 0081PE - 0088PE
education	Less than 9th grade 9th to 12th grade, no diploma High school graduate (includes equivalency) Some college, no degree Associate's degree Bachelor's degree Graduate or professional degree	DP02, 0059PE - 0065PE

Table 3.1: Information on American Community Survey distributional variables. For each variable X, we show its support as well as the associated ACS variable codes from `https://api.census.gov/data/2018/acs/acs5/profile/variables.html`.

Chapter 4.

Belief Propagation for Networks with Loops

In Chapters 2 and 3, we looked at how to characterize the structure of networks with metadata on the nodes and/or edges, discussing specific applications in signed networks and spatial networks with distribution-valued metadata. We now move onto the second primary goal of this thesis: to develop scalable, accurate, and interpretable inference techniques for real-world network data. In this chapter, we will examine how to make efficient inferences in models where the underlying network has many short loops, a case which is not typically handled well by simple approximations like mean-field or standard belief propagation (see Appendix A for details on belief propagation in the context of SBM inference). By developing a new belief propagation method that accounts for the highly clustered structure of real networks and is computationally efficient, we open up the possibility for statistical inference of a variety of probabilistic models on large-scale real-world network data.

4.1. Introduction

Phenomena of interest are often modeled using probabilistic formulations that capture the probabilities of states of network nodes. Examples include the spread of epidemics through networks of social contacts [190], cascading failures in power grids [191], and the equilibrium behavior of spin models such as the Ising model [192]. Networks are also used to represent pairwise dependencies between variables in statistical models that do not otherwise have a network component, as a convenient tool for bookkeeping and visualization of model structure [193]. Such "graphical models," which allow us to represent the conditional dependencies between variables in a non-parametric manner, form the foundation for many modern machine learning techniques [194]. Belief propagation can also be used independently of an explicit graphical model, for example in machine learning tasks such as node classification [195, 196], for which it is a popular technique due to its computational efficiency.

The solution of probabilistic models like this presents a challenge. Analytic methods such as those used for regular lattices do not generalize to the more complex topologies of networks, and mean-field and other standard approximations often fail to take crucial details of network structure into account. Numerical methods can be successful but are computationally demanding on larger networks and sometimes give results of poor accuracy. Message passing or "belief propagation" methods offer an alternative and promising approach that straddles the line between analytic and numerical techniques [197, 198]. Message passing works by deriving a set of self-consistent equations satisfied by the variables or probabilities of interest and then solving those equations by numerical iteration. The name "message passing" comes from the fact that the equations can be thought of as representing messages passed

between neighboring nodes in the network.

Standard formulations of message passing, however, have a crucial weakness: they rely on the assumption that the states of the neighbors are uncorrelated with one another, which is only true if the network contains no loops. Unfortunately, almost all real-world networks do contain loops, and usually many of them [199], so standard message passing can give quite poor results in practical situations. In this chapter we propose a solution to this problem in the form of a new class of message passing methods for probabilistic models on "loopy" networks. These methods open up a host of possibilities for novel network calculations, many of which we discuss here.

The limitations of traditional message passing have been widely noted in the past and a number of previous attempts have been made to remedy them. The only truly loopless networks are trees, but standard message passing methods have been shown to give good results on networks that satisfy the weaker condition of being "locally tree-like," meaning that local regions of the network take the form of trees even though the network as a whole is not a tree. In effect, this means that the network can contain long loops, but not short ones [1]. However, realistic networks often fail to satisfy even this weaker condition and contain many short loops. Message passing has been extended to certain classes of random graphs with short loops, such as Husimi graphs [200–202] and other tree-like agglomerations of small loopy subgraphs [203, 204], but these techniques are not generally applicable to real-world networks. Alternatively, one can incorporate the effect of loops by using a perturbative expansion around the loopless case [205, 206], though this approach becomes progressively less accurate as the number of loops increases and is therefore best suited to networks with a low loop density, which rules out a large fraction of real networks, whose loop density is often high [199, 207].

Perhaps the best known extension of belief propagation, and the one most similar to our own approach, is the method known as generalized belief propagation [208], which is based on the idea of passing messages not just between pairs of nodes but between larger groups. Generalized belief propagation employs a *region-based approximation* [209], in which the free energy $\ln Z$ is approximated by a sum of independent local free energies of regions within the network. Once the regions are defined it is straightforward to write down belief propagation equations, which can be used to calculate marginals and other quantities of interest, including approximations to the partition function and entropy. Perhaps the best known example of generalized belief propagation, at least within the statistical physics community, is the *cluster variational method*, in which the regions are defined so as to be closed under the intersection operation [210] and the resulting free energy is called the *Kikuchi free energy* [211].

The accuracy and complexity of generalized belief propagation is determined by the specific choice of regions, which has been described as being "more of an art than a science" [212]. Loops contained within regions are correctly accounted for in the belief propagation, while those that span two or more regions are not and introduce error. At the same time, the computational complexity of the belief propagation calculations increases exponentially with the size of the regions [212], so choosing the right regions is a balancing act between enclosing as many loops as possible while not making the regions too large. A number of heuristics have been proposed for choosing the regions [213–215] but real-world networks can pose substantial difficulties because they often contain both high degrees and many loops [1], which effectively forces us to compromise either by leaving loops out or by using very large regions. Our method can have a significant advantage in these systems because it can accommodate large, tightly connected neighborhoods through local Monte Carlo sampling.

Our method also has the benefit that the neighborhoods are constructed automatically based on the network structure rather than being chosen by the user.

In Ref. [216] message passing schemes are described for percolation models and spectral calculations on loopy networks. In this study we extend this approach to the solution of general probabilistic models. We derive a factorization of the probability of states for such models that allows us to write self-consistent message passing equations for the marginal probabilities on sets of nodes in a neighborhood around a given reference node. From these equations we can then calculate a range of quantities of interest such as single-site marginals, partition functions, and entropies. To ground our discussion we use the Ising model as an example of our approach, showing how our improved message passing methods can produce better estimates for this model than regular message passing. We show that our methods are asymptotically exact on networks whose loop structure satisfies certain general conditions and give good approximations for networks that deviate from these conditions. We give example results for the Ising model on both real and artificial networks and also discuss applications of our method to a range of other problems, emphasizing its wide applicability.

4.2. Methods

Our first step is to develop the general theory of message passing for probabilistic models on loopy networks. With an eye on the Ising model, our discussion will be in the language of spin models, although the methods we describe can be applied to any probabilistic model with pairwise dependencies between variables, making it suitable for a broad range of calculations in probabilistic modeling.

4.2.1. Model description

Consider a general undirected, unweighted network G composed of a set V of nodes or vertices and a set E of pairwise edges. As discussed in Sec. 1.1, the network can be represented mathematically by its adjacency matrix $\mathbf{A}$ with elements $A_{ij} = 1$ when nodes i and j are connected by an edge and 0 otherwise. On each node of the network there is a variable or spin s_i, which is restricted to some discrete set of values S. In a compartmental model of disease propagation, for instance, $s_i \in S = \{0\ (\text{susceptible}), 1\ (\text{infected}),\ 2\ (\text{removed})\}$ could be the infection state of a node [217, 218]. In a spatial model of segregation $s_i \in S = \{0\ (\text{unoccupied}),\ 1\ (\text{occupied})\}$ could represent land occupation [219].

Spins s_i and s_j interact if and only if there is an edge between nodes i and j, a formulation sufficiently general to describe a large number of models in fields as diverse as statistical physics, machine learning, economics, psychology, epidemiology, and sociology [210, 220–225]. Interactions are represented by an interaction energy $g_{ij}(s_i, s_j|\omega_{ij})$, which controls the preference for any particular pair of states s_i and s_j to occur together. The quantity ω_{ij} represents any external parameters, such as temperature in a classical spin system or infection rate in an epidemiological model, that control the nature of the interaction. We also allow for the inclusion of an external field $f_i(s_i|\theta_i)$ with parameters θ_i, which controls the intrinsic propensity for s_i to take an particular state. This could be used for instance to encode individual risk of catching a disease in an epidemic model.

Given these definitions, we write the probability $P(\mathbf{s}|\omega, \theta)$ that the complete set of

spins takes value $\mathbf{s}$ in the Boltzmann form

$$P(\mathbf{s}|\omega,\theta) = \frac{e^{-H(\mathbf{s}|\omega,\theta)}}{Z(\omega,\theta)}, \tag{4.1}$$

where the Hamiltonian

$$H(\mathbf{s}|\omega,\theta) = -\sum_{(i,j)\in E} g_{ij}(s_i, s_j|\omega_{ij}) - \sum_{i\in V} f_i(s_i|\theta_i) \tag{4.2}$$

is the log-probability of the state to within an arbitrary additive constant, and the partition function

$$Z(\omega,\theta) = \sum_{\mathbf{s}} e^{-H(\mathbf{s}|\omega,\theta)} \tag{4.3}$$

is the appropriate normalizing constant, ensuring that $P(\mathbf{s}|\omega,\theta)$ sums to unity. In this chapter we will primarily be concerned with computing the single-site (or one-point) marginal probabilities

$$q_i(s_i) = \sum_{\mathbf{s}\setminus s_i} P(\mathbf{s}|\omega,\theta), \tag{4.4}$$

where $\mathbf{s} \setminus s_i$ denotes all spins with the exception of s_i. For convenience we have dropped ω and θ from the notation on the left of the equation, but it should be clear contextually that q_i depends on both of these variables.

The one-point marginals reveal useful information about physical systems, such as the magnetization of classical spin models or the position of a phase transition. They are important for statistical inference problems, where they give the posterior probability of a variable taking a given state after averaging over contributions from

all other variables (e.g., the total probability of an individual being infected with a disease at a given time). Unfortunately, direct computation of one-point marginals is difficult because the number of terms in the sum in Eq. (4.4) grows exponentially with the number of spins. The message passing method gives us a way to get around this difficulty and compute q_i accurately and rapidly.

Message passing can also be used to calculate other quantities. For instance, we will show how to compute the average energy (also called the internal energy), which is given by

$$U(\omega, \theta) = \sum_{\mathbf{s}} H(\mathbf{s}|\omega, \theta) P(\mathbf{s}|\omega, \theta). \tag{4.5}$$

The average energy is primarily of interest in thermodynamic calculations, although it may also be of interest for statistical inference, where it corresponds to the average log-likelihood.

We can also compute the two-point correlation function between spins

$$P(s_i = x, s_j = y) = P(s_j = y|s_i = x)\, q_i(s_i = x). \tag{4.6}$$

This function can be computed by first calculating the one-point marginal $q_i(s_i = x)$, then fixing $s_i = x$ and repeating the calculation for s_j. The same approach can also be used to compute higher order multi-point correlation functions.

4.2.2. Message passing equations

Our method operates by dividing a network into neighborhoods [216]. A neighborhood $N_i^{(r)}$ around node i is defined as the node i itself and all of its edges and

neighboring nodes, plus all nodes and edges along paths of length r or less between the neighbors of i. See Fig. 4.1 for examples. The key to our approach is to focus initially on networks in which there are no paths longer than r between the neighbors of i, meaning that all paths are inside $N_i^{(r)}$. This means that all correlations between spins within $N_i^{(r)}$ are accounted for by edges that are also within $N_i^{(r)}$, which allows us to write exact message passing equations for these networks. Equivalently, we can define a *primitive cycle* of length r starting at node i to be a cycle (i.e., a self-avoiding loop) such that at least one edge in the cycle is not on any shorter cycle beginning and ending at i. Our methods are then exact on any network that contains no primitive cycles of length greater than $r + 2$.

This approach gives us a series of methods where the rth member of the series is exact on networks that contain primitive cycles of length $r+2$ and less only. The calculations become progressively more complex as r gets larger: they are very tractable for smaller values but become impractical when r is large. In many real-world networks the longest primitive loop will be relatively long, requiring an infeasible computation to reach an exact solution. However, long loops introduce smaller correlations between variables than short ones, and moreover the density of long loops is in many cases low: the network is "locally dense but globally sparse." In this situation, we find that the message passing equations for low values of r, while not exact, give excellent results. They account correctly for the effect of the short loops in the network, while making only a small approximation by omitting the long ones.

In practice, quite modest values of r can give good results. The smallest possible choice is $r = 0$, which corresponds to assuming that there are no loops in the network at all, that the network is a tree. This is the assumption made by traditional message passing methods, and it gives poor results on many real-world networks. The next

approximation after this, however, with $r = 1$, which correctly accounts for the effect of loops of length three in the network (i.e., triangles), produces substantially better results, and the $r = 2$ approximation (which includes loops of length three and four) is in many cases impressively accurate. In the following developments, we drop r from our notation for convenience—the same equations apply for all values of r.

Having defined the initial neighborhood N_i we further define a neighborhood $N_{j\setminus i}$ to be node j plus all edges in N_j that are not contained in N_i and the nodes at their ends. Our method involves writing the marginal probability distribution on the spin at node i in terms of a set of messages received from nodes j that are in N_i, including nodes that are not immediate neighbors of i. (This contrasts with traditional message passing in which messages are received only from the immediate neighbors of i.) These messages are then in turn calculated from further messages j receives from nodes $k \in N_{j\setminus i}$, and so forth.

When written in this manner, the messages i receives are independent of one another in any network with no primitive cycles longer than $r + 2$. Messages received from any two nodes j_1 and j_2 within N_i are necessarily independent since they are calculated from the corresponding neighborhoods $N_{j_1\setminus i}$ and $N_{j_2\setminus i}$ which are disconnected from one another: if they were connected by any path then that path would create a primitive cycle starting at i but passing outside of N_i, of which by hypothesis there are none. By the same argument, we also know that $N_{j\setminus i}$ and N_i only overlap at the single node j for any $j \in N_i$.

This much is in common with the approach in Ref. [216], but to apply these ideas to the solution of probabilistic models we need to go further. Specifically, we now show how this neighborhood decomposition allows us to factorize the Hamiltonian into a product of independent sums over the individual neighborhoods, with interactions

that can be represented by messages passed between neighborhoods. Consider N_i as comprising a central set of nodes and edges surrounding i. Then we can think of the set of neighborhoods $N_{j\setminus i}$ for all $j \in N_i$ as comprising the next "layer" in the network, the sets $N_{k\setminus j}$ for all $k \in N_{j\setminus i}$ as a third layer, and so forth until all nodes and edges in the network are accounted for. In a network with no primitive cycles longer than $r+2$, this procedure counts all interactions exactly once, allowing us to rewrite our Hamiltonian as a sum of independent contributions from the various layers thus:

$$\begin{aligned} H(\mathbf{s}) = H_{N_i}(\mathbf{s}_{N_i}) &+ \sum_{j\in N_i} H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}}) \\ &+ \sum_{j\in N_i}\sum_{k\in N_{j\setminus i}} H_{N_{k\setminus j}}(\mathbf{s}_{N_{k\setminus j}}) \\ &+ \sum_{j\in N_i}\sum_{k\in N_{j\setminus i}}\sum_{l\in N_{k\setminus j}} H_{N_{l\setminus k}}(\mathbf{s}_{N_{l\setminus k}}) + \dots, \end{aligned} \tag{4.7}$$

where $\mathbf{s}_{N_i}$ and $\mathbf{s}_{N_{j\setminus i}}$ are the sets of spins for the nodes in the neighborhoods N_i and $N_{j\setminus i}$ and we have defined the local Hamiltonians

$$H_{N_i}(\mathbf{s}_{N_i}) = -\sum_{(j,k)\in N_i} g_{jk}(s_j, s_k|\omega_{jk}) - f_i(s_i|\theta_i), \tag{4.8}$$

$$H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}}) = -\sum_{(k,l)\in N_{j\setminus i}} g_{kl}(s_k, s_l|\omega_{kl}) - f_j(s_j|\theta_j). \tag{4.9}$$

The decomposition of Eq. (4.7) is illustrated pictorially in Fig. 4.1.

The essential feature of this decomposition is that it breaks sums over spins such as those in Eqs. (4.3) and (4.4) into a product of sums over the individual neighborhoods $\{N_{j\setminus i}\}_{j\in N_i}$. Because these neighborhoods are, as we have said, independent, this means that the partition function and related quantities factorize into products of

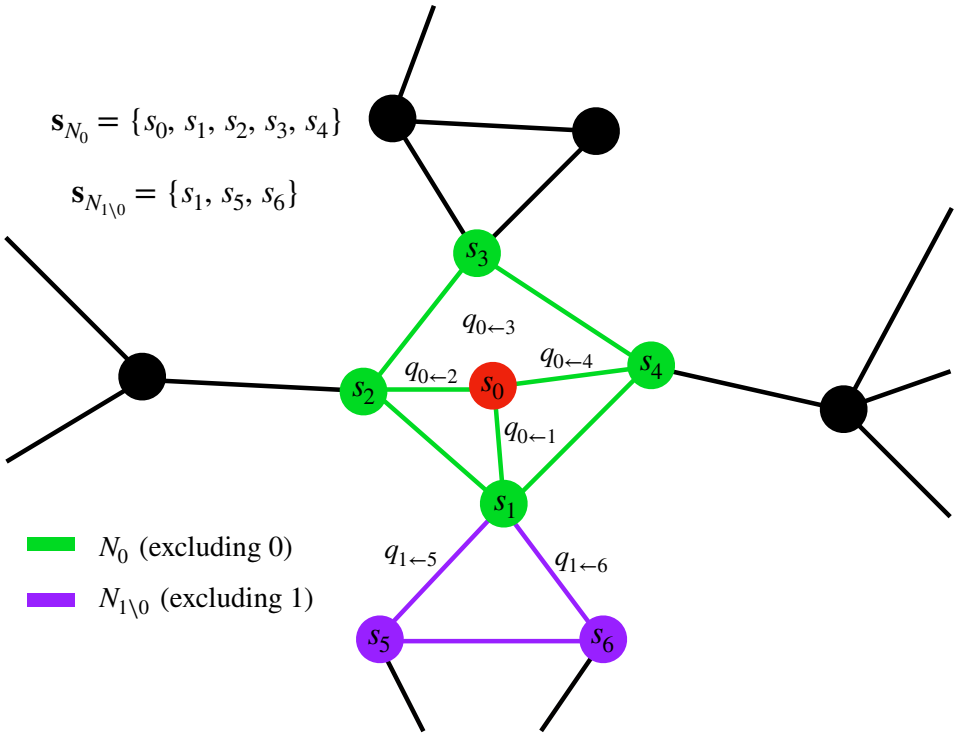

Fig. 4.1. Hamiltonian expansion diagram, with $r = 2$. The focal node is in red while the rest of its neighborhood N_0 is in green. Nodes and edges in purple represent the neighborhood $N_{1\setminus 0}$ excluding node 1. We also label the corresponding spin and message variables used in Eqs. (4.11) and (4.12).

sums over a few spins each, which can easily be performed numerically. For instance, the one-point marginal of Eq. (4.4) takes the form

$$q_i(s_i = x) \propto \sum_{\mathbf{s}_{N_i} : s_i = x} e^{-H_{N_i}(\mathbf{s}_{N_i})} \prod_{j \in N_i} \sum_{\mathbf{s}_{N_{j\setminus i}\setminus j}} e^{-H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})} \prod_{k \in N_{j\setminus i}} \sum_{\mathbf{s}_{N_{k\setminus j}\setminus k}} e^{-H_{N_{k\setminus j}}(\mathbf{s}_{N_{k\setminus j}})} \dots, \tag{4.10}$$

which can be written recursively as

$$q_i(s_i = x) = \frac{1}{Z_i} \sum_{\mathbf{s}_{N_i} : s_i = x} e^{-H_{N_i}(\mathbf{s}_{N_i})} \prod_{j \in N_i} q_{i\leftarrow j}(s_j), \tag{4.11}$$

with

$$q_{i\leftarrow j}(s_j = y) = \frac{1}{Z_{i\leftarrow j}} \sum_{\mathbf{s}_{N_{j\setminus i}} : s_j = y} e^{-H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})} \prod_{k \in N_{j\setminus i}\setminus j} q_{j\leftarrow k}(s_k), \tag{4.12}$$

where the normalization constants Z_i and $Z_{i\leftarrow j}$ ensure that the marginals q_i and messages $q_{i\leftarrow j}$ are normalized so that they sum to 1. (In practice, we simply normalize the messages by dividing by their sum.) The quantity $q_{i\leftarrow j}(s_j)$ is equal to the marginal probability of node j having spin s_j when all the edges in N_i are removed. Alternatively, one can think of it as a local external field on node j that influences the probability distribution of s_j. To make this more explicit one could rewrite Eq. (4.11) as

$$q_i(s_i = x) = \frac{1}{Z_i} \sum_{\mathbf{s}_{N_i} : s_i = x} e^{-H_{N_i}(\mathbf{s}_{N_i}) + \sum_{j \in N_i} \log q_{i\leftarrow j}(s_j)}, \tag{4.13}$$

where $\log q_{i\leftarrow j}(s_j)$ plays the role of the external field.

Equations (4.11) and (4.12) define our message passing algorithm and can be solved for the messages $q_{i\leftarrow j}$ by simple iteration, starting from any suitable set of starting values and applying the equations repeatedly until convergence is reached.

With only slight modification we can use the same approach to calculate the internal energy as well. The contribution to the internal energy from the interactions of a single node i is $\frac{1}{2}\sum_{j:A_{ij}=1} g(s_i, s_j|\omega_{ij}) + f(s_i|\theta_i)$, where the factor of $\frac{1}{2}$ compensates for double counting of interactions. Summing over all nodes i and weighting by the appropriate Boltzmann probabilities, the total internal energy is

$$U = \sum_{i \in V} \frac{1}{Z_i} \sum_{\mathbf{s}_{N_i}} \left[\frac{1}{2} \sum_{j:A_{ij}=1} g(s_i, s_j|\omega_{ij}) + f(s_i|\theta_i) \right] e^{-H_{N_i}(\mathbf{s}_{N_i})} \prod_{j \in N_i} q_{i\leftarrow j}(s_j). \tag{4.14}$$

All of the quantities appearing here are known a priori, except for the messages $q_{i \leftarrow j}(s_j)$ and the normalizing constants Z_i, which are calculated in the message passing process. Performing the message passing and then using the final converged values in Eq. (4.14) then gives us our internal energy.

4.2.3. Implementation

For less dense networks, those with node degrees up to about 20, the message passing equations of Eqs. (4.11) and (4.12) can be implemented directly and work well. The method is also easily parallelizable, as we can update all messages asynchronously based on their values from the previous iteration, as well as compute the final marginals in parallel.

For networks with higher degrees the calculations can become unwieldy, the huge reduction in complexity due to the factorization of the Hamiltonian notwithstanding. For a model with t distinct spin states at every node, the sum over states in the neighborhood of i has $t^{|N_i|}$ terms, which can quickly become computationally expensive to evaluate. Moreover, if just a single node has too large a neighborhood it can make the entire computation intractable, as that single neighborhood can consume more computational power than is available.

In such situations, therefore, we take a different approach. We note that Eq. (4.12) is effectively an expectation

$$q_{i \leftarrow j}(s_j = y) = \langle \delta_{s_j, y} \rangle_{N_{j \setminus i}}, \tag{4.15}$$

where we use the shorthand

$$\langle A \rangle_{N_{j \setminus i}} = \sum_{\mathbf{s}_{N_{j \setminus i}}} A(\mathbf{s}_{N_{j \setminus i}}) \frac{e^{-H_{N_{j \setminus i}}(\mathbf{s}_{N_{j \setminus i}})} \prod_{k \in N_{j \setminus i} \setminus j} q_{j \leftarrow k}(s_k)}{Z_{i \leftarrow j}}. \tag{4.16}$$

We approximate this average using Markov chain Monte Carlo importance sampling over spin states, and after convergence of the messages the final estimates of the marginals q_i can also be obtained by Monte Carlo, this time on the spins in N_i. We describe the details in Section 4.3.

4.2.4. Calculating the partition function

The partition function Z is perhaps the most fundamental quantity in equilibrium statistical mechanics. From a knowledge of the partition function one can calculate virtually any other thermodynamic variable of interest. Objects equivalent to Z also appear in other fields, such as Bayesian statistics, where the quantity known as the *model evidence*, the marginal likelihood of observed data given a hypothesized model, is mathematically analogous to the partition function and plays an important role in model fitting and selection [226–228].

Unfortunately, the partition function is difficult to calculate in practice. The calculation can be done analytically in some special cases [229, 230], but direct numerical calculations are difficult due to the need to sum over an exponentially large number of states, and Monte Carlo is challenging because of the difficulty of correctly normalizing the Boltzmann distribution.

Another concept central to statistical mechanics is the entropy

$$S = -\sum_{\mathbf{s}} P(\mathbf{s}) \ln P(\mathbf{s}), \tag{4.17}$$

which has broad applications not just in physics but across the sciences [231–233]. Like the partition function, entropy is difficult to calculate numerically, and for exactly the same reasons, since the two are closely related. For the canonical distribution of Eq. (4.1) the entropy is given in terms of Z by $S = \ln Z + \beta U$. Even if we know the internal energy U therefore (which is relatively straightforward to compute), the entropy is at least as hard to calculate as the partition function. Indeed the fundamental difficulty of normalizing the Boltzmann distribution is equivalent to establishing the zero of the entropy, a well known hard problem (unsolvable within classical thermodynamics, requiring the additional axiom of the Third Law).

As we now show, the entropy can be calculated using our message passing formalism by appropriately factorizing the probability distribution over spin states. Since we have already developed a prescription for computing U (see Eq. (4.14)), this also allows us to calculate the partition function. The details of the procedure are quite involved and do not follow straightforwardly from the previous discussion, so we defer the derivation to Appendix B.4. As shown there, the state probability $P(\mathbf{s})$ in Eq. (4.1) can be rewritten in the factorized form

$$P(\mathbf{s}) = \frac{\prod_{i \in G} P(\mathbf{s}_{N_i})}{\prod_{((i,j)) \in G} P(\mathbf{s}_{\cap_{ij}})^{2/|\cap_{ij}|}}, \tag{4.18}$$

where $P(\mathbf{s}_{N_i})$ is the joint marginal distribution of the variables in the neighborhood of node i, $P(\mathbf{s}_{\cap_{ij}})$ is the joint marginal distribution in the intersection $\cap_{ij} = N_i \cap N_j$ of

the neighborhoods N_i and N_j, and $((i,j))$ denotes pairs of nodes that are contained in each other's neighborhoods.

By a series of manipulations, this form can be further expressed as the pure product

$$P(\mathbf{s}) = \left[\prod_{((i,j))\in G} P(\mathbf{s}_{\cap_{ij}})^{1/\binom{|\cap_{ij}|}{2}}\right]\left[\prod_{(i,j)\in G} P(s_i,s_j)^{W_{ij}}\right]\left[\prod_{i\in G} P(s_i)^{C_i}\right], \tag{4.19}$$

where

$$W_{ij} = 1 - \sum_{((l,m))\in G} \frac{1}{\binom{|\cap_{lm}|}{2}} \mathbf{1}_{\{(i,j)\in\cap_{lm}\}} \tag{4.20}$$

with $\mathbf{1}_{\{\dots\}}$ being the indicator function, and

$$C_i = 1 - \left(\sum_{j\in N_i} \frac{1}{|\cap_{ij}| - 1}\right) - \left(\sum_{j\in N_i^{(0)}} W_{ij}\right). \tag{4.21}$$

Substituting Eq. (4.19) into Eq. (4.17), we get an expression for the entropy thus:

$$\begin{aligned} S = &-\frac{1}{\binom{|\cap_{ij}|}{2}} \sum_{((i,j))\in G} P(\mathbf{s}_{\cap_{ij}}) \ln P(\mathbf{s}_{\cap_{ij}}) \\ &- \sum_{(i,j)\in G} W_{ij} P(s_i,s_j) \ln P(s_i,s_j) - \sum_{i\in G} C_i P(s_i) \ln P(s_i). \end{aligned} \tag{4.22}$$

Note that, like the well known Bethe approximation for the entropy [212], this expression has contributions from the one- and two-point marginals $P(s_i)$ and $P(s_i,s_j)$ of Eqs. (4.6) and (4.11), but also contains a term that depends on the joint marginal $P(\mathbf{s}_{\cap_{ij}})$ in the intersection $\cap_{ij}$, which may be nontrivial if $r > 0$. As shown in Appendix B.4,

we can calculate this joint marginal using the message passing equation

$$P(\mathbf{s}_{\cap_{ij}}) = \frac{1}{Z_{\cap_{ij}}} e^{-\beta H(\mathbf{s}_{\cap_{ij}})} q_{i \leftarrow j}(s_j) \prod_{k \in \cap_{ij} \setminus j} q_{j \leftarrow k}(s_k), \tag{4.23}$$

where $H(\mathbf{s}_{\cap_{ij}})$ denotes the terms of the Hamiltonian of Eq. (4.2) that fall within $\cap_{ij}$ and $Z_{\cap_{ij}}$ is the corresponding normalizing constant. For $|\cap_{ij}|$ sufficiently small, $Z_{\cap_{ij}}$ can be computed exactly. In other cases we can calculate it using Monte Carlo methods similar to those we used previously for the marginals $P(s_i)$.

4.2.5. Ising model calculations

As an archetypal application of our methods we consider the Ising model on various example networks. The ferromagnetic Ising model in zero external field is equivalent in our notation to the choices

$$g_{ij}(s_i, s_j) = -\beta A_{ij} s_i s_j, \qquad f_i(s_i) = 0, \tag{4.24}$$

where $\beta = 1/T$ is the inverse temperature. Note that temperature in this notation is considered a part of the Hamiltonian. It is more conventional to write temperature separately, so that the Hamiltonian has dimensions of energy rather than being dimensionless as here, but absorbing the temperature into the Hamiltonian is notationally convenient in the present case. It effectively makes the temperature a parameter ω_{ij} in Eq. (4.2) (and all ω_{ij} are equal).

As example calculations, we will compute the average magnetization M, which is

given by

$$M = \left| \left\langle \frac{1}{N} \sum_{i=1}^{N} s_i \right\rangle \right| = \frac{1}{N} \left| \sum_{i=1}^{N} [2q_i(s_i = +1) - 1] \right|, \tag{4.25}$$

and the heat capacity C, given by

$$C = \frac{dU}{dT} = -\beta^2 \frac{dU}{d\beta}. \tag{4.26}$$

As detailed in Appendix B.1, we employ an extension of the message passing equations to compute C that avoids having to use a numerical derivative to evaluate Eq. (4.26). In brief, we consider the messages $q_{i \leftarrow j}$ to be a function of β then define their derivatives with respect to β as their own set of messages

$$\eta_{i \leftarrow j} = \frac{dq_{i \leftarrow j}}{d\beta}, \tag{4.27}$$

with their own associated message passing equations derived by differentiating Eq. (4.12). We then compute the heat capacity C by differentiating Eq. (4.14), expressing the result in terms of the $\eta_{i \leftarrow j}$, and substituting it into Eq. (4.26).

4.2.6. Behavior at the phase transition

In many geometries, the ferromagnetic Ising model has a phase transition at a nonzero critical temperature between a symmetric state with zero average magnetization and a symmetry broken state with nonzero magnetization. Substituting Eq. (4.24) into Eqs. (4.11) and (4.12) we can show that the message passing equations for the Ising model always have a trivial solution $q_{i \leftarrow j}(s_j) = \frac{1}{2}$ for all i, j. This choice is a fixed point

of the message passing iteration: when started at this point the iteration will remain there indefinitely. Looking at Eq. (4.25), we see that this fixed point corresponds to magnetization $M = 0$. If the message passing iteration converges to this trivial fixed point, therefore, it tells us that the magnetization is zero and we are above the critical temperature; if it settles elsewhere then the magnetization is nonzero and we are below the critical temperature. Thus the phase transition corresponds to the point at which the fixed point changes from being attracting to being repelling.

This behavior is well known in standard belief propagation, where it has been shown that on networks with long loops only there is a critical temperature T_{BP} below which the trivial fixed point becomes unstable and hence the system develops nonzero magnetization, and that this temperature corresponds precisely to the conventional zero-field continuous phase transition on these networks [234]. Extending the same idea to the present case, we expect the phase transition on a loopy network to fall at the corresponding transition point between stable and unstable in our message passing formulation.

Moreover, because the values of the messages at the trivial fixed point are known, we can compute an expression for the phase transition point without performing any message passing. We treat the message passing iteration as a dynamical system and perform a linear stability analysis of the trivial fixed point. Perturbing around $q = \frac{1}{2}$ (shorthand for setting all $q_{i\leftarrow j} = \frac{1}{2}$) and keeping terms to linear order, we find that the dynamics is governed by the Jacobian

$$J_{j\to i,\nu\to\mu} = \frac{\partial q_{i\leftarrow j}}{\partial q_{\mu\leftarrow\nu}}\Big|_{q=1/2} = \tilde{B}_{j\to i,\nu\to\mu} D_{j\to i,\nu\to\mu}, \tag{4.28}$$

where $\tilde{B}$ is a generalization of the so-called non-backtracking matrix [235] to our loopy

message passing formulation:

$$\tilde{B}_{j\to i,\nu\to\mu} = \begin{cases} 1 & \text{if } j=\mu \text{ and } \nu \in N_{j\setminus i}, \\ 0 & \text{otherwise,} \end{cases} \tag{4.29}$$

and $D_{j\to i,\nu\to\mu}$ is a correlation function between the spins s_μ and s_ν within the neighborhood $N_{j\setminus i}$—see Appendix B.3 for details.

When the magnitude of the leading eigenvalue $\lambda_{\max}$ of this Jacobian is less than 1, the trivial fixed point is stable; when it is greater than 1 the fixed point is unstable. Hence we can locate the phase transition temperature by numerically evaluating the Jacobian and locating the point at which $|\lambda_{\max}|$ crosses 1, for instance by binary search.

Equation 4.29 is also useful in its own right. The non-backtracking matrix has numerous applications within network science, for instance in community detection [235], centrality measures [236], and percolation theory [237]. The generalization defined in Eq. 4.29 could be used to extend these applications to loopy networks, although we will not explore such calculations here.

4.3. Results

4.3.1. A model network

As a first example application, we examine the behavior of our method on a model network created precisely to have short loops only up to a specified maximum length. The network has short primitive cycles only of length $r+2$ and less for a given choice of r, though it can also have long loops—it is "locally dense but globally sparse" in the sense discussed previously. Indeed this turns out to be a crucial point. The Ising

model does not have a normal phase transition on a true tree, because at any finite temperature there is always a nonzero density of defects in the spin state (pairs of adjacent spins that are oppositely oriented), which on a tree divide the network into finite sized regions, imposing a finite correlation length and hence no critical behavior. Similarly in the case of a network with only short loops and no long ones there is no true phase transition. The long loops are necessary to produce criticality, a point discussed in detail in [238].

To generate networks that have short primitive cycles only up to a certain length, we first generate a random bipartite network—a network with two types of nodes and connections only between unlike kinds, as discussed in Sec. 1.1—then project down onto one type of node, producing a network composed of a set of complete subgraphs or cliques. In detail, the procedure is as follows.

1. We first specify the degrees of all the nodes, of both types, in the bipartite network.

2. We represent these degrees by stubs of edges emerging, in the appropriate numbers, from each node, then we match stubs at random in pairs to create our random bipartite network.

3. We project this network onto the nodes of type 1, meaning that any two such nodes that are both connected to the same neighbor of type 2 are connected directly with an edge in the projection. After all such edges have been added, the type 2 nodes are discarded.

4. Finally, we remove a fraction p of the edges in the projected network at random.

If $p = 0$, the network is composed of fully connected cliques, but when $p > 0$ some

cliques will be lacking some edges, and hence the network is composed of a collection of subgraphs of size equal to the degrees of the corresponding nodes of type 2 from which they were projected. If we limit these degrees to a maximum value of $r+2$ then there will be no short loops of length longer than this.

Figure 4.2 shows the magnetization per spin, entropy, and heat capacity for the ferromagnetic Ising model on an example network of $9\,447$ nodes and $13\,508$ edges generated using this procedure with $r = 2$ and $p = 0.6$. We also limit the degrees of the type-1 nodes in the bipartite graph to a maximum of 5, which ensures that no neighborhood in the projection is too large to prevent a complete summation over states and hence that Monte Carlo estimation of the sums in the message passing equations is unnecessary.

Results are shown for belief propagation calculations with $r = 0$, 1, and 2, the last of which should, in principle, be exact except for the weak correlations introduced by the presence of long loops in the network. We also show in the figure the magnitude of the leading eigenvalue of J for each value of r. The points at which this eigenvalue equals 1, which give estimates of the critical temperature for each r, are indicated by the vertical lines. Also shown in the figure for comparison are results from direct Monte Carlo simulations of the system, with the entropy calculated from values of the heat capacity computed from energy fluctuations and then numerically integrated using the identity

$$S = \int_0^T \frac{C(T)}{T}\, dT. \tag{4.30}$$

The message passing simulations offer significantly faster results for this system: for $r = 2$ message passing was about 100 times faster than the Monte Carlo simulations.

Looking at Fig. 4.2, we can see that as we increase r the message passing results approach those from the direct Monte Carlo, except close to the phase transition, where the Monte Carlo calculations suffer from finite size effects that smear the phase transition, to which the message passing approach appears largely immune. While the results for conventional belief propagation ($r = 0$) are quite far from the direct Monte Carlo results, most of the improvement in accuracy from our method is already present even at $r = 1$. Going to $r = 2$ offers only a small additional improvement in this case.

The apparent position of the phase transition aligns well with the predictions derived from the value of the Jacobian for each value of r. The transition is particularly clear in the gradient discontinuity of the magnetization. For $r = 1$ and 2 the heat capacity appears to exhibit a discontinuity at the transition, which differs from the divergence we expect on low-dimensional lattices but bears a resemblance to the behavior seen on Bethe lattices and other homogeneous tree-like networks [197, 239, 240].

4.3.2. Real-world networks

For our next example we look at an application on a real-world network, where we do not expect the method to be exact, though as we will see it nonetheless performs well. The network we examine has larger local neighborhoods than our synthetic example, which means we are not able to sum exhaustively over all configurations of the spins $\mathbf{s}_{N_{j\setminus i}}$ in Eq. (4.12) (and similarly $\mathbf{s}_{N_i}$ in Eq. (4.11)) so, as described in Section 4.2.3, we instead make use of Monte Carlo sampling to estimate the messages $q_{i\leftarrow j}$ and marginals q_i.

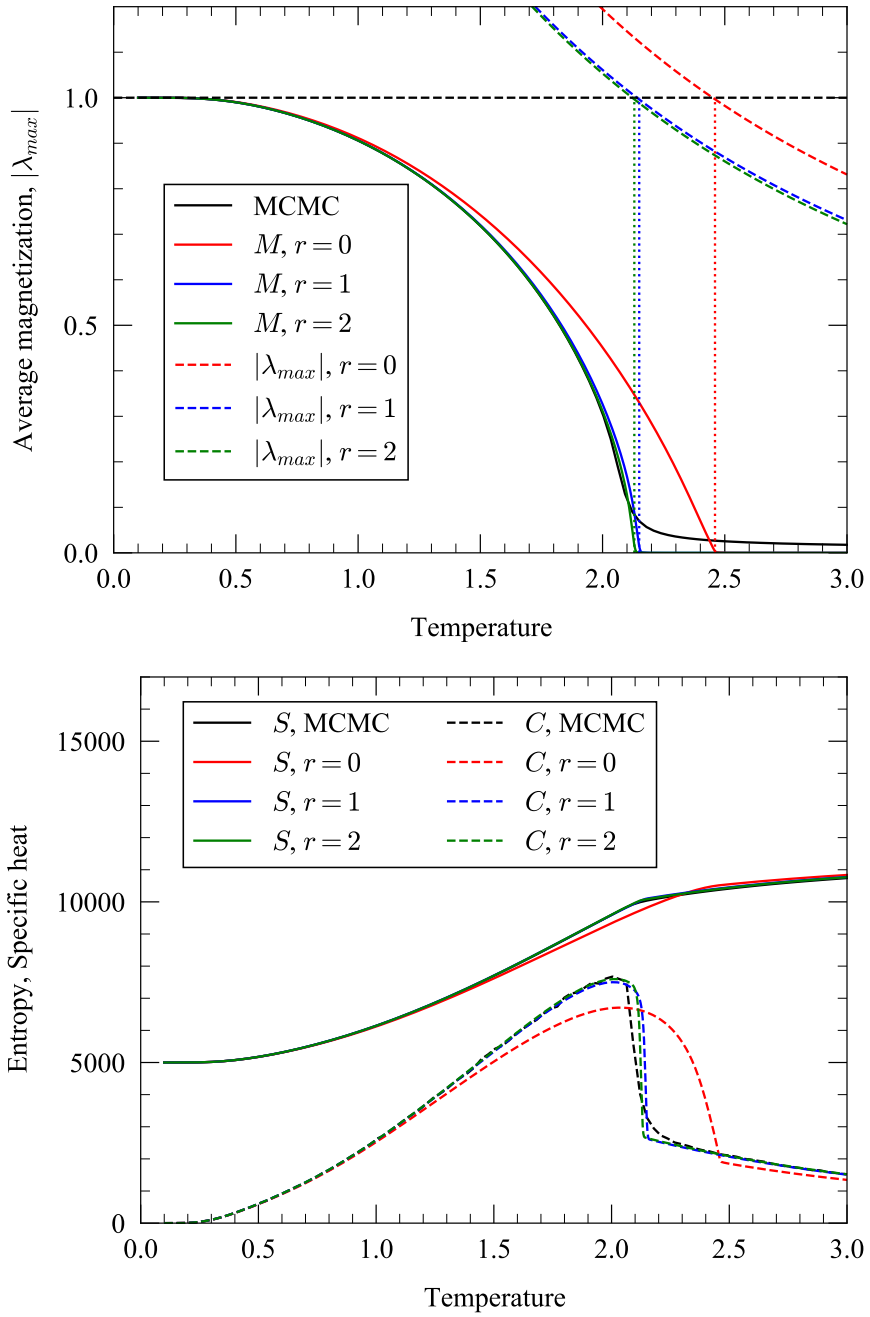

Fig. 4.2. Ferromagnetic Ising model critical behavior on synthetic network. The top panel shows the average magnetization, while the bottom one shows the heat capacity and the entropy (the latter shifted up for visualization purposes). The magnitude of the leading eigenvalue for the Jacobian is also shown in the top panel for all three values of r, and we can see that the apparent positions of the phase transition, revealed by discontinuities in the physical quantities or their gradients, correspond closely to the temperatures at which the associated eigenvalues are equal to 1.

The summation over local spins in Eq. (4.12) is equivalent to computing the expectation in Eq. (4.15). To calculate $q_{i\leftarrow j}(s_j = y)$ we fix the values of its incoming messages $\{q_{j\leftarrow k}\}$ and perform Monte Carlo sampling over the states of the spins in the neighborhood $N_{j\setminus i}$ with the Hamiltonian of Eq. (4.9). Then we compute the average in Eq. (4.15) separately for the cases $y = 1$ and -1 and normalize to ensure that the results sum to one. The resulting values for $q_{i\leftarrow j}$ can then be used as incoming messages for calculating other messages in other neighborhoods. We perform the Monte Carlo using the Wolff cluster algorithm [241], which makes use of the Fortuin-Kasteleyn percolation representation of the Ising model to flip large clusters of spins simultaneously and can significantly reduce the time needed to obtain independent samples, particularly close to the critical point. Once the messages have converged to their final values we compute the marginals q_i by performing a second Monte Carlo, this time over the spins $\mathbf{s}_{N_i}$ with the Hamiltonian of Eq. (4.8). More details on the procedure are given in Appendix B.2.

The Monte Carlo approach combines the best aspects of message passing and traditional Monte Carlo calculations. Message passing reduces the sums we need to perform to sets of spins much smaller than the entire network, while the Monte Carlo approach dramatically reduces the number of spin *states* that need to be evaluated. The approach has other advantages too. For instance, because of the small neighborhood sizes it shows improved performance in systems with substantial energy barriers that might otherwise impede ergodicity, such as antiferromagnetic systems. But perhaps its biggest advantage is that it effectively allows us to sample very large numbers of states of the network without taking very large samples of individual neighborhoods. If we sample k configurations from one neighborhood and k configurations from another, then in effect we are summing over k^2 possible combinations of states in the

union of the two neighborhoods. Depending on the value of r, there are at least $2m$ neighborhoods $N_{j \setminus i}$ in a network, where m is the number of edges, and hence we are effectively summing over at least k^{2m} states overall, a number that increases exponentially with network size. Effective sample sizes of 10^{1000} or more are easily reachable, far beyond what is possible with traditional Monte Carlo methods.

Figure 4.3 shows the results of applying these methods with $r = 0 \ldots 4$ to a network from [242] representing the structure of an electric power grid, along with results from direct Monte Carlo simulations on the same network. As the figure shows, the magnetization is again poorly approximated by the traditional ($r = 0$) message passing algorithm, but improves as r increases. In particular, the behavior in the region of the phase transition is quite poor for $r = 0$ and does not provide a good estimate of the position of the transition. For $r = 1$ and 2, however, we get much better estimates, and for $r = 3$ and 4 the method approaches the Monte Carlo results quite closely, with the critical temperature falling somewhere in the region of $T = 1.6$ in this case. We also see a much clearer phase transition in the message passing results than in the standard Monte Carlo, because of finite size effects in the latter. These results all suggest that for real systems our method can give substantial improvements over both ordinary belief propagation and direct Monte Carlo simulation, and in some cases show completely different behavior altogether.

4.4. Conclusion

In this chapter we have presented a new class of message passing algorithms for solving probabilistic models on networks that contain a high density of short loops. Taking the Ising model as an example, we have shown that our methods give sub-

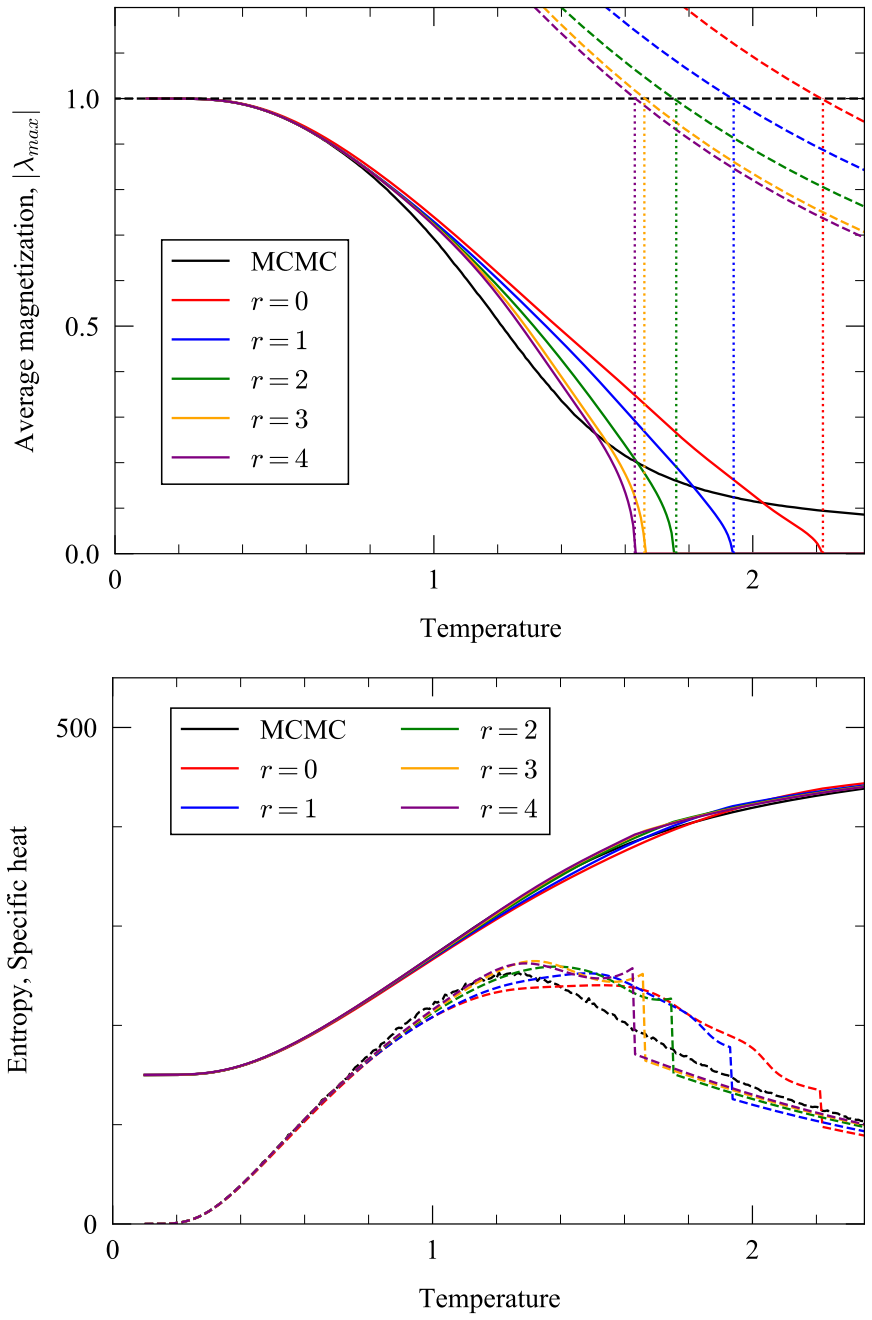

Fig. 4.3. Ferromagnetic Ising model critical behavior on a power grid network. Message passing and Monte Carlo calculations of the average magnetization, entropy, and specific heat on the "494 bus power system" network from Ref. [242]. Again, the message passing results approximate the real solution progressively better as r grows larger.

stantially improved results in calculations of magnetization, heat capacity, entropy, marginal spin probabilities, and other quantities over standard message passing methods that do not account for the presence of loops. Our methods are exact on networks with short loops up to a fixed maximum length which we can choose, and can give good approximations on networks with loops of any length.

Chapter 5.

Representative Community Divisions of Networks

In this chapter, we conclude our discussion of statistical inference for network data by addressing the issue of multimodality in community detection. We describe a simple, fast, principled method for dealing with the heterogeneity present in the results from any community detection algorithm, revealing the range of possibilities for community structure in a network in a manner analogous to standard error analysis for measurement data. This method allows researchers employing community detection in their work to summarize the complex output of these algorithms to gain a more holistic understanding of the ways in which a network can be effectively partitioned.

5.1. Introduction

There are numerous existing methods for community detection, including ones based on centrality measures [243], modularity [244], information theory [245], and

Bayesian generative models [246]—see [247] for a review. Most methods represent the community structure in a network as a single network partition or division (see Sec. 1.2.4 for a brief background), which is typically the one that attains the highest score according to some objective function. As pointed out by many previous authors, however, there may be multiple partitions of a network that achieve high scores, any of which could be a good candidate for division of the network [248–253]. With this in mind some community detection methods, including methods based on modularity and on generative models, return multiple plausible partitions rather than just one. But while these algorithms give a more complete picture of community structure, they have their own problems. In particular, the number of partitions returned is often very large. Even for relatively small networks the partitions may number in the hundreds or thousands, far more than any human observer can reasonably comprehend. How then are we supposed to make sense of the output of these calculations?

In some cases it may happen that all of the plausible divisions of a network are quite similar to each other, in which case we may be able to form a *consensus clustering* [254], a single partition that is representative of the entire set in the same way that the mean of a set of numbers can be a useful representation of the whole. However, if the partitions vary substantially, then the consensus can fail to capture the full range of behaviors in the same way that the mean can be a poor summary statistic for broad or multimodal distributions of numbers. In cases like these, summarizing the community structure may require not just one but several representative partitions, which may themselves be consensus partitions for a local cluster of network divisions [253]. In this chapter, we present a simple and efficient method for finding such representative partitions. Given a large set of possible structures returned by a community detection algorithm, our method finds a smaller set that captures the main variants

and possibilities while remaining comprehensible to human users.

Broadly speaking, our method clusters the partitions into a small number of subsets, in a manner somewhat akin to traditional methods for clustering numerical data in high-dimensional data spaces. A few previous studies have investigated the clustering of partitions. Calatayud et al. [255] proposed an algorithm that starts with the single highest scoring partition (under whatever objective function is in use), then iterates through other divisions in order of decreasing score and assigns each to the closest cluster if the distance to that cluster is less than a certain threshold, or starts a new cluster otherwise. This approach is primarily applicable in situations where there is a clear definition of distance between partitions (there are many possible choices [256]), as the results turn out to be sensitive to this definition and to the corresponding distance threshold. Peixoto [253] has proposed a principled statistical method for clustering partitions using methods of Bayesian inference, which works well but differs from ours in that rather than returning a single partition as a representative of each cluster it returns a distribution over partitions. It also does not explicitly address issues of the dependence of the number of clusters on the number of input partitions, issues that we address in some detail in this chapter.

The method we propose is based on fundamental information theoretic principles and has a number of practical advantages. It does not require the explicit choice of partition distance function, does not depend on the number of input partitions provided the partition space is well sampled, and is adaptable to any community detection algorithm that returns multiple sample partitions. The method is based on the principle of *minimum description length*, which posits that when selecting between possible models for a data set, the best model is the one that permits the most succinct representation of the data [257]. In our context, we seek to capture the information

contained in a set of community divisions returned by some community detection algorithm using a model that consists of a small number of representative partitions that are used to reconstruct the clusters around them. The description length principle has been applied to clustering in the past for real-valued (non-network) data, including methods based on Gaussian mixture models [258], hierarchical clustering [259], Bernoulli mixture models for categorical data [260], and probabilistic generative models [261]. Georgieva et al. [262], for instance, have proposed a clustering framework that is similar in some respects to ours but for real-valued vector data. As in our approach the data are thought of as a message to be transmitted in multiple parts, including the cluster centers and the data within each cluster. Georgieva et al., however, only use their measure as a quality function to assess the outputs of other clustering algorithms and not as an objective to be optimized to obtain the clusters themselves. The minimum description length approach has also been applied to the task of community detection itself by Rosvall and Bergstrom [263], who used it to formulate an objective function for community detection that considers the encoding of a network in terms of a partition and the node and edge counts within and between the communities in the partition.

Our algorithm takes as input a set of divisions of a network into communities, which may be obtained in any manner we like. Common methods for generating such divisions are sampling from probabilistic models, thermal samples generated using modularity or other energy functions, or multiple runs of optimization algorithms, and our method will work with any of these. We design a partition clustering objective function using simple information theoretic arguments, and use an efficient Monte Carlo scheme to optimize this objective and identify clusters of similar partitions and a representative member of each cluster. We test the method on a range

of real and synthetic networks and demonstrate that it returns substantially distinct community divisions that are a good guide to the structures present in the original sample.

5.2. Methods

The primary goal of our proposed technique is to find representative partitions that summarize the community structure in a network. We call these representative partitions *modes*. Suppose we have an observed network consisting of n nodes and we have some method for finding community divisions of these nodes, also called partitions. As in Sec. 1.2.4, we can represent a partition with a length-n vector $\mathbf{g}$ that assigns to each node $i = 1 \ldots n$ a label g_i indicating which community it belongs to.

We assume that there are a large number of plausible partitions and that our community detection method returns a subset of them. Normally we expect that many of the partitions would be similar to one another, differing only by a few nodes here or there. The goal of this study is to develop a procedure for gathering such similar partitions into clusters, and generating a mode, which is itself a partition, as an archetypal representative of each cluster. For the sake of clarity, we will use the words "partition" or "division" to describe the assignment of network nodes to communities, and the word "cluster" to describe the assignment of entire partitions to groups according to the method that we describe.

In order both to divide the partitions into clusters and to find a representative mode for each cluster, we first develop a clustering objective function based on information theoretic arguments. The main concept behind our approach is a thought experiment in which we imagine transmitting our set of partitions to a receiver using a multi-

step encoding chosen so as to minimize the amount of information required for the complete transmission.

5.2.1. Partition clustering as an encoding problem

Let us denote our set of partitions by D and suppose there are S partitions in the set, labeled $p = 1 \ldots S$. Now imagine we wish to transmit a complete description of all elements of the set to a friend. How should we go about this? The most obvious way is to send each of the partitions separately to the receiver using some simple encoding that uses, say, numbers or symbols to represent community labels. We could do somewhat better by using an optimal prefix code such as a Huffman code [264] that economizes by representing frequently used labels with shorter code words. Even this, however, would be quite inefficient in terms of information. We can do better by making use of the fact that, as we have said, we expect many of our partitions to be similar to one another. This allows us to save information by dividing the partitions into clusters of similar ones and transmitting only a few partitions in full—one representative partition or mode for each cluster—then describing the remaining partitions by how they differ from these modes. The method is illustrated in Fig. 5.1.

Initially, let us assume that we want to divide the set D of partitions into K clusters, denoted C_k with $k = 1 \ldots K$. (We will discuss how to choose K separately in a moment.) To efficiently transmit D, we first transmit K representative modes, which themselves are members of D, with group labels $\hat{\mathbf{g}}^{(k)}$. Then for each individual partition in D we transmit which cluster, or equivalently which mode, it belongs to and then the partition itself by describing how it differs from that mode. Since the latter information will be smaller if a partition is more similar to its assigned mode, choosing

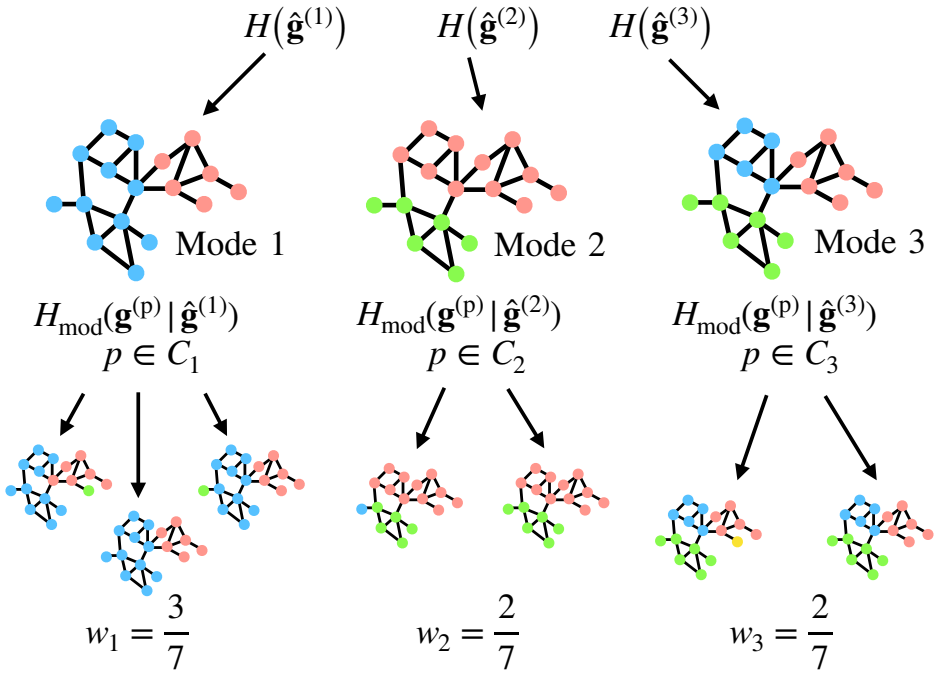

Fig. 5.1. Illustration of the transmission of a set of partitions for a network. We first transmit a small set of "modes," archetypal partitions drawn from the larger set, with average information content equal to the entropies of these partitions (Eq. 5.2). Then each partition p from the complete set is transmitted by describing how it differs from the most similar of the modes, using a modified conditional entropy expression (Eq. 5.4). The weight w_k is the fraction of all partitions that are part of cluster k.

a set of modes that are accurately representative of all partitions will naturally minimize the total information, and we use this criterion to derive the best set of modes. This is the minimum description length principle, as applied to finding the optimal clusters and modes.

Following this plan, the total description length per sampled partition can be ap-

proximated (see Appendix C.1) by the expression

$$\mathcal{L}_{\text{total}} = \frac{n}{S}\sum_{k=1}^{K} H(\hat{\mathbf{g}}^{(k)}) + H(\mathbf{c}) + \frac{n}{S}\sum_{k=1}^{K}\sum_{p\in C_k} H_{\text{mod}}(\mathbf{g}^{(p)}|\hat{\mathbf{g}}^{(k)}). \tag{5.1}$$

The first term represents the amount of information required to transmit the modes and is simply equal to the sum of their entropies:

$$H(\hat{\mathbf{g}}^{(k)}) = -\sum_{r=1}^{n_{m_k}} \frac{a_r^{(m_k)}}{n} \log \frac{a_r^{(m_k)}}{n}. \tag{5.2}$$

Here m_k is the partition label p of the kth mode, n_p is the number of communities in partition p, and $a_r^{(p)}$ is the number of nodes in partition p that have community label r.

The second term in Eq. 5.1 represents the amount of information needed to specify which cluster, or alternatively which mode, each partition in D belongs to:

$$H(\mathbf{c}) = -\sum_{k=1}^{K} \frac{c_k}{S} \log \frac{c_k}{S}, \tag{5.3}$$

where $c_k = |C_k|$ is the number of partitions (out of S total) that belong to mode k.

The third term in 5.1 represents the amount of information needed to specify each of the individual partitions $\mathbf{g}^{(p)}$ in terms of their modes $\hat{\mathbf{g}}^{(k)}$:

$$H_{\text{mod}}(\mathbf{g}^{(p)}|\hat{\mathbf{g}}^{(k)}) = H(\mathbf{g}^{(p)}|\hat{\mathbf{g}}^{(k)}) + \frac{1}{n}\log \Omega(p, m_k). \tag{5.4}$$

H_{mod} is the *modified conditional entropy* of the group labels of $\mathbf{g}^{(p)}$ given the group labels

of $\hat{\mathbf{g}}^{(k)}$ [265]. The normal (non-modified) conditional entropy is

$$H(\mathbf{g}^{(p)}|\hat{\mathbf{g}}^{(k)}) = -\sum_{r=1}^{n_{m_k}} \sum_{s=1}^{n_p} \frac{t_{rs}^{m_k p}}{n} \log \frac{t_{rs}^{m_k p}}{a_r^{(m_k)}}, \tag{5.5}$$

where t_{rs}^{mp} is the number of nodes simultaneously classified into community r in partition $\mathbf{g}^{(m)}$ and community s in partition $\mathbf{g}^{(p)}$. The matrix of elements $\mathbf{t}^{mp}$ for any pair of partitions m, p is known as a *contingency table*, and Eq. 5.5 measures the amount of information needed to transmit $\mathbf{g}^{(p)}$ given that we already know both $\hat{\mathbf{g}}^{(k)}$ *and* the contingency table. To actually transmit the partitions in practice we would also need to transmit the contingency table, and the second term in Eq. 5.4 represents the information needed to do this. The quantity $\Omega(p, m)$ is equal to the number of possible contingency tables $\mathbf{t}^{mp}$ with row and column sums $a_r^{(m)}$ and $a_s^{(p)}$ respectively. This quantity can be computed exactly for smaller contingency tables and there exist good approximations to its value for larger tables [265].

The modified conditional entropy, including the $\log \Omega$ term, thus measures the total amount of information needed to transmit the partition $\mathbf{g}^{(p)}$ after having already transmitted its mode $\hat{\mathbf{g}}^{(k)}$. The $\log \Omega$ term is often omitted from calculations of conditional entropy, but it turns out to be crucial in the current application. Without it, one can minimize the conditional entropy simply by making the number of groups in the modal partition very large, with the result that the minimum description length solution is biased toward modes with many groups. The additional term avoids this bias.

In principle, before we send any of this information, we also need transmit to the receiver information about the size of each partition and the number of modes K, which adds some additional terms to the description length, Eq. 5.1. These terms, however,

are small, and moreover they are independent of how we configure our clusters and modes, so we can safely neglect them.

A detailed derivation of Eq. 5.1 is given in Appendix C.1. By minimizing this quantity we can now find the best set of modes to describe a given set of partitions.

5.2.2. Choosing the number of clusters

So far we have assumed that we know the number K of clusters of partitions, or equivalently the number of modes. In practice we do not usually know K and normally there is not even one "correct" value for a given network. Different values of K can give useful answers for the same network, depending on how much granularity we wish to see in the community structure. In general, a small numbers of clusters—no more than a dozen or so—is most informative to human eyes, but fewer clusters also means that each cluster will contain a wider range of structures within it. How then do we choose the value of K?

One might hope for a principled, parameter-free method of choosing the value based for instance on statistical model selection techniques, in which we allow the data to dictate the natural number of clusters that should be used to describe it. On closer inspection, however, it seems likely that no such method exists. As described for example in [253], one can look at the problem in terms of the "landscape" defined by a community detection metric such as modularity. Good community structures correspond to high values of modularity and clusters of good structures correspond to regions of high value or peaks in the landscape, so the number of clusters is equivalent to the number of peaks. But the number of peaks depends on how closely one inspects the landscape. Viewed at a coarse scale the landscape may contain only a

few peaks, but at a finer scale there will be many small fluctuations that define large numbers of peaks and valleys, each potentially corresponding to its own cluster. One must make a decision about the scale at which one wishes to probe the landscape and this decision is equivalent to choosing the number of clusters. This is not a question of avoiding "overfitting," as can happen in certain types of model fitting. The landscape here is a deterministic one and the fluctuations are not due to stochastic variation or measurement error. They would persist even if we could draw an infinite number of sample structures for a network.

Thus, if one wants to generate a reasonable number of clusters one is obliged to make a decision about what that number is and devise a way to impose that decision on the output of the clustering algorithm. There are no "parameter-free" ways to perform the clustering. Some methods might appear at first glance to be parameter-free, but this only means that the parameters are concealed in implementation details or in assumptions made in the algorithm design. In our approach we prefer to make the parameters explicit for the user, both to clarify the assumptions made in a calculation and to give the user the option to vary the parameter values if they wish.

A natural way to parametrize the number of clusters is to impose a penalty on the description length objective function using a multiplier or "chemical potential" that couples linearly to the value of K thus:

$$\mathcal{L}_{\text{total}} = \frac{n}{S}\sum_{k=1}^{K} H(\hat{\mathbf{g}}^{(k)}) + H(\mathbf{c}) + \frac{n}{S}\sum_{k=1}^{K}\sum_{p\in C_k} H_{\text{mod}}(\mathbf{g}^{(p)}|\hat{\mathbf{g}}^{(k)}) + \lambda K. \tag{5.6}$$

This imposes a penalty equal to λ for each extra cluster added and hence larger values of λ will produce larger penalties.

Equation (5.6) is the objective function we use in our method. It is straightforward

to show that this form makes the optimal number of clusters K independent of S—see Appendix C.2 for a derivation. As we have said, we normally want to the number of modes to be small, which means that we expect λ to be of order unity. In practice, we find that the choice $\lambda = 1$ works well in many cases and this is the value we use for all the example applications presented here, although it is possible that other values might be useful in certain circumstances.

One can set the value of λ to zero, which is equivalent to removing the penalty term altogether. In this case there is still an optimal choice of K implied by the description length alone. Low values of K, corresponding to only a small number of modes, will give inefficient descriptions of the data because many partitions will not be similar to any of the modes, while high values of K will give inefficient partitions because we will waste a lot of information describing all the modes. In between, at some moderate value of K, there is an optimal choice that determines the best number of clusters. An analogous method is used, for example, for choosing the optimal number of bins for histograms and often works well in that context [266, 267].

This might appear at first sight to give a parameter-free approach to choosing the number of modes, but in fact this is not the case because the number of modes the method returns now depends on the number of sampled partitions S, increasing as the value of S increases and diverging as S becomes arbitrarily large. When creating a histogram from a fixed set of samples this behavior is desirable—you want to use more bins when you have more data—but in the present case the dependence on S introduces a hidden parameter on which the value of K depends. We would like our representation of the space of community structures to capture the fundamental features of the network independent on how we choose to sample those features, including how many samples we draw.

It is worth noting that one can envisage other encodings of a set of community structures that would give slightly different values for the description length. For example, when transmitting information about which cluster each sampled structure belongs to one could choose to use a single fixed-length code for the cluster labels, which would require $\log K$ bits per sample. This would simply replace the term $H(\mathbf{c})$ in Eq. 5.1 with $\log K$. One could analogously replace the terms $H(\hat{\mathbf{g}}^{(k)})$ with their corresponding fixed-length average code sizes (per node), with values $\log n_{m_k}$. In general, both of these changes would result in a less efficient encoding that tends to favor a smaller number of modes. However, neither of them would affect the asymptotic scaling of the description length and the term in λK would still be needed to achieve a number of modes that is independent of S. It is also possible to extend the description length formulation to a hierarchical model in which we allow the possibility of more than one "level" of modes being transmitted, which could compress the data more efficiently but lacks the simple interpretation of the output present in the two-level scheme presented here.

5.2.3. Minimizing the clustering objective

Our goal is now to find the set of modes $\hat{\mathbf{g}}$ that minimize Eq. 5.6. This could be done using any of a variety of optimization methods, but here we make use of a greedy algorithm that employs a sequence of elementary moves that merge and split clusters, inspired by a similar merge-split algorithm for sampling community structures described in [268]. We start by randomly dividing our set D of partitions into some number K_0 of initial clusters, then identify the mode $\hat{\mathbf{g}}^{(k)}$ of each cluster C_k as the partition $p \in C_k$ that minimizes $H(\mathbf{g}^{(p)}) + \sum_{q \in C_k} H_{\mathrm{mod}}(\mathbf{g}^{(q)}|\mathbf{g}^{(p)})$. In other words, the

initial mode for each cluster is the partition p that is closest to all other partitions q in the cluster in terms of modified conditional entropy, accounting for the entropy of p itself.

Computing the modified conditional entropy, Eq. 5.4, has time complexity $\mathrm{O}(n)$, which means it takes $\mathrm{O}(nS^2/K_0^2)$ steps to compute each mode exactly if the initial clusters are the same size. This can be slow in practice, but we can obtain a good approximation substantially faster by Monte Carlo sampling. We draw a random sample X of partitions from the cluster (without replacement) and then minimize $H(\mathbf{g}^{(p)}) + (c_k/|X|) \sum_{q \in X} H_{\mathrm{mod}}(\mathbf{g}^{(q)}|\mathbf{g}^{(p)})$, where as previously c_k is the size of the cluster. Good results can be obtained with relatively small samples, and in our calculations we use $|X| = 30$. The time complexity of this calculation is $\mathrm{O}(nS/K_0)$, a significant improvement given that sample sizes S can run into the thousands or more. We also store the values of $H(\mathbf{g}^{(p)})$ and $H_{\mathrm{mod}}(\mathbf{g}^{(q)}|\mathbf{g}^{(p)})$ as they are computed so that they do not need to be recomputed on subsequent steps of the algorithm.

Technically, our formulation does not require one to constrain $\hat{\mathbf{g}}^{(k)}$ to be a member of C_k, but this restriction significantly reduces the computation time in practice by allowing stored conditional entropy values to be reused repeatedly during calculation. One could relax this restriction and choose the mode $\hat{\mathbf{g}}^{(k)}$ of each cluster C_k to be the partition $\mathbf{g}$ (which may or may not be in C_k) that minimizes $H(\mathbf{g}) + \sum_{q \in C_k} H_{\mathrm{mod}}(\mathbf{g}^{(q)}|\mathbf{g})$. However, we have not taken this approach in the examples presented here.

Once we have an initial set of clusters and representative modes, the algorithm proceeds by repeatedly proposing one of the following moves at random, accepting it only if it reduces the value of Eq. 5.6:

1. Pick a partition $\mathbf{g}^{(p)}$ at random and assign it to the closest mode $\hat{\mathbf{g}}^{(k)}$, in terms of modified conditional entropy.

2. Pick two clusters $C_{k'}$ and $C_{k''}$ at random and merge them into a single cluster C_k, recomputing the cluster mode as before.

3. Pick a cluster C_k at random and split it into two clusters $C_{k'}$ and $C_{k''}$ using a k-means style algorithm: we select two modes at random from C_k and assign each partition in C_k to the closer of the two (in terms of modified conditional entropy). Then we recompute the modes for each resulting cluster and repeat until convergence is reached.

These steps together constitute a complete algorithm for minimizing Eq. 5.6 and optimizing the clusters, but we find that the efficiency of the algorithm can be further improved by adding a fourth move:

4. Perform step 2, then immediately perform step 3 using the merged cluster from step 2.

This extra move, inspired by a similar one in the community merge-split algorithm of [268], helps with the rapid optimization of partition assignments between pairs of clusters.

We continue performing these moves until a prescribed number of consecutive moves are rejected without improving the objective function. We find that this procedure returns very consistent results despite its random nature. If results were found to vary between runs it could be worthwhile to perform random restarts of the algorithm and adopt the results with the lowest objective score. However, this has not proved necessary for the examples presented here.

The algorithm has $\mathrm{O}(nS)$ time complexity per move in the worst case (which occurs when there is just a single cluster), and is fast in practice. In particular, it is typically

much faster than the community detection procedure itself for current community detection algorithms, so it adds little to the overall time needed to analyze a network. We give a range of example applications in the next section.

5.3. Results

In this section we demonstrate the application of our method to a number of example networks, both real and computer generated. For each example we perform community detection by fitting to the non-parametric degree-corrected block model [269] and sampling $10\,000$ community partitions from the posterior distribution of the model by Markov chain Monte Carlo using the algorithm of [268]. These samples are then clustered using the method of this chapter with the cluster penalty parameter set to $\lambda = 1$, the number of Monte Carlo samples for estimating modes to $|X| = 30$, and the number of initial modes to $K_0 = 1$. We also calculate for each mode k a weight $w_k = c_k/S$ equal to the fraction of all partitions in D that fall in cluster k, to assess the relative sizes of the clusters.

5.3.1. Synthetic networks

As a first test of our method, we apply it to a set of synthetic (i.e., computer-generated) networks specifically constructed to display varying degrees of ambiguity in their community structure. Figure 5.2A shows results for a network generated using the planted partition model, a symmetric version of the stochastic block model [67, 77] in which n nodes are assigned in equal numbers to q communities, and between each pair of nodes i, j an edge is placed with probability p_{in} if i and j are in the same community or p_{out} if i and j are in different communities. In our example we generated a

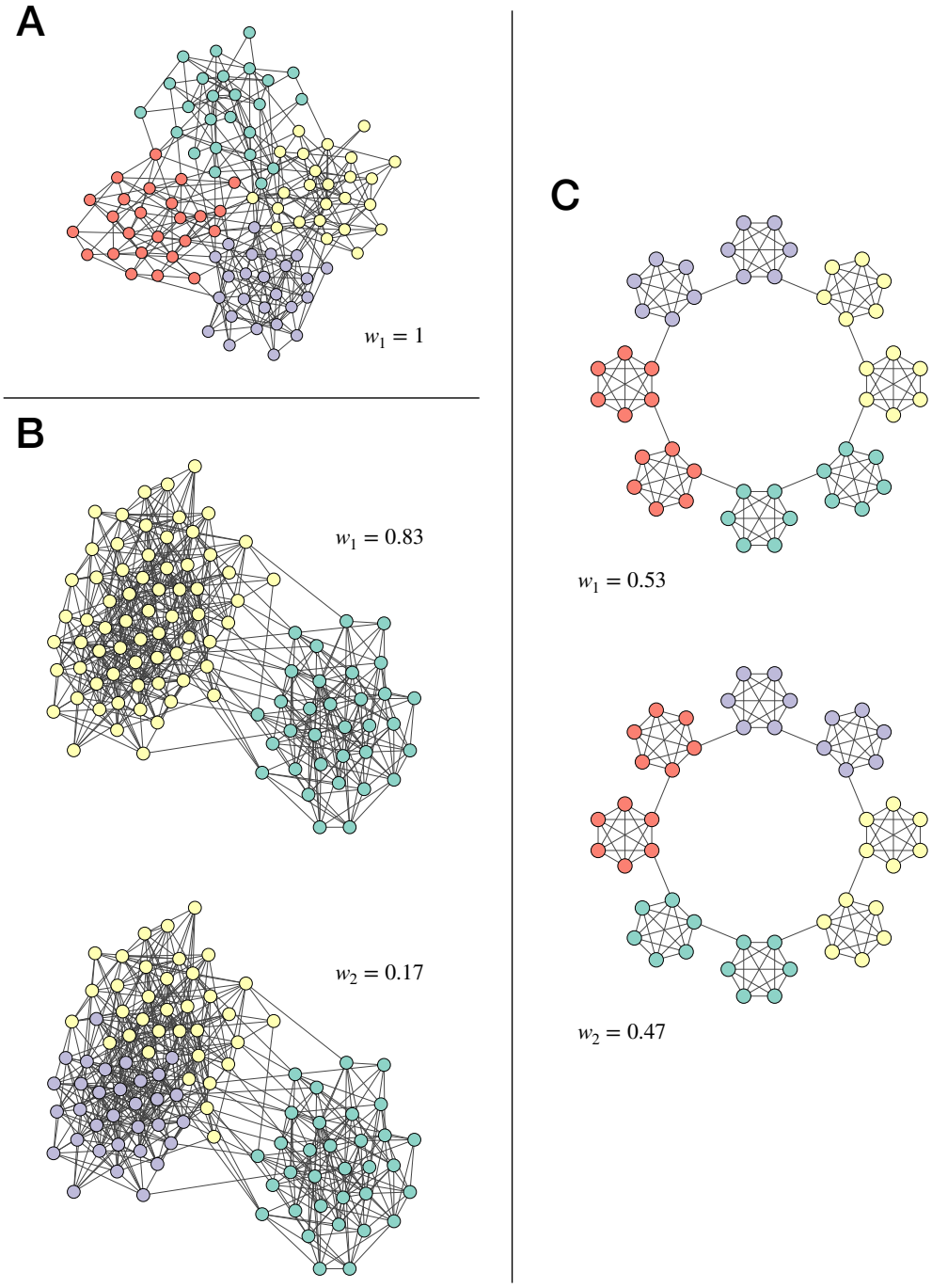

Fig. 5.2. Representative modes and their corresponding weights for three synthetic example networks, identified by minimizing Eq. 5.6 with $\lambda = 1$ for $10,000$ community partition samples. (A) Planted partition model with 100 nodes, four communities, and connection probabilities $p_{\text{in}} = 0.25$ and $p_{\text{out}} = 0.02$. (B) Network of 99 nodes generated using the stochastic block model with a mixing matrix of the form given in Eq. 5.7 with $p_s = 0.27$, $p_m = 0.08$, and $p_b = 0.01$. (C) Ring of eight cliques of six nodes each, connected by single edges, based on the example in [270].

network with $n = 100$ nodes, $q = 4$ communities, and $p_{\text{in}} = 0.25$, $p_{\text{out}} = 0.02$. Though it contains four communities, by its definition, this network should exhibit only a single mode, the structure "planted" into it in the network generation process. There will be competing individual partitions, but they should be distributed evenly around the single modal structure rather than multimodally around two or more structures. And indeed our algorithm correctly infers this as shown in the figure: it returns a single representative structure in which all nodes are grouped correctly into their planted communities. Given the random nature of the community detection algorithm it would be possible for a small number of nodes to be incorrectly assigned in the modal structure, simply by chance, but in the present case this did not happen and every node is assigned correctly.

For a second, more demanding example we construct a network using the full (non-symmetric) stochastic block model, which is more flexible than the planted partition model. If $\mathbf{g}$ denotes a vector of community assignments as previously, then an edge in the model is placed between each node pair i, j independently at random with probability $\omega_{g_i g_j}$, where the $\omega_{g_i g_j}$ are parameters that we choose (see Sec. 1.3.1). For our example we create a network with three communities and with parameters of the form

$$\boldsymbol{\omega} = \begin{bmatrix} p_s & p_m & p_b \\ p_m & p_s & p_b \\ p_b & p_b & p_s \end{bmatrix}, \tag{5.7}$$

where p_s is the within-group edge probability, p_m and p_b are between-group probabilities, and $p_s > p_m > p_b$. In our particular example the network has $N = 99$ nodes divided evenly between the three groups and $p_s = 0.27$, $p_m = 0.08$, $p_b = 0.01$.

This gives the network a nested structure in which there is a clear separation between group 3 and the rest, and a weaker separation between groups 1 and 2. This sets up a deliberate ambiguity in the community structure: does the "correct" structure have three groups or just two? As shown in Fig. 5.2B, our method accurately pinpoints this ambiguity, finding two representative modes for the network, one with three separate communities and one where communities 1 and 2 are merged together.

A third synthetic example network is shown in Fig. 5.2C, the "ring of cliques" network of Fortunato and Barthelemy [270], in which a set of cliques (i.e., complete subgraphs) are joined together by single edges to create a loop. Good et al. [251] found this network to have ambiguous community structure in which the cliques joined together in pairs rather than forming separate communities on their own. Since there are two symmetry-equivalent ways to divide the ring into clique pairs this also means there are two equally good divisions of the network into communities. Good et al. performed their community detection using modularity maximization, but similar behavior is seen with the method used here. Most sampled community structures show the same division into pairs of cliques, except for a clique or two that may get randomly assigned as a whole to a different community. Our algorithm readily picks out this structure as shown in Fig. 5.2C, finding two modes that correspond to the two rotationally equivalent configurations. Moreover, the two modes have approximately equal weight w_k in the sampling, indicating that the Monte Carlo algorithm spent a roughly equal amount of time on partitions near each mode.

5.3.2. Real networks

Turning now to real-world networks, we show that our method can also accurately summarize community structure found in a range of practical domains. (Further examples are given in Appendix C.3.) The results demonstrate not only that the method works but also that real-world networks commonly do have multimodal community structure that is best summarized by two or more modes rather than by just a single consensus partition, although our method will return a single partition when it is justified—see Sec. 5.3.1.

Figure 5.3A shows results for one well-studied network, the co-purchasing network of books about politics compiled by Krebs (unpublished, but see [59]), where two books are connected by an edge if they were frequently purchased by the same buyers. It has been conjectured that this network contains two primary communities, corresponding to politically left- and right-leaning books, but the network contains more subtle divisions as well. A study by Peixoto [253] found 11 different types of structure—what we are here calling "modes." Many of these modes, however, differed only slightly, by the reassignment of a few nodes from one community to another. Applying our method to the network we find, by contrast, just two modes as shown in the figure, suggesting that our algorithm is penalizing minor variations in structure more heavily than that of Ref. [253]. The two modes we find have four communities each. In the one on the left in Fig. 5.3A these appear to correspond approximately to books that are politically liberal (red), center-left (purple), center-right (green), and conservative (yellow); in the one on the right they are left-liberal (green), liberal (red), center (purple), and conservative (yellow).

Figure 5.3B shows a different kind of example, a social network of self-reported

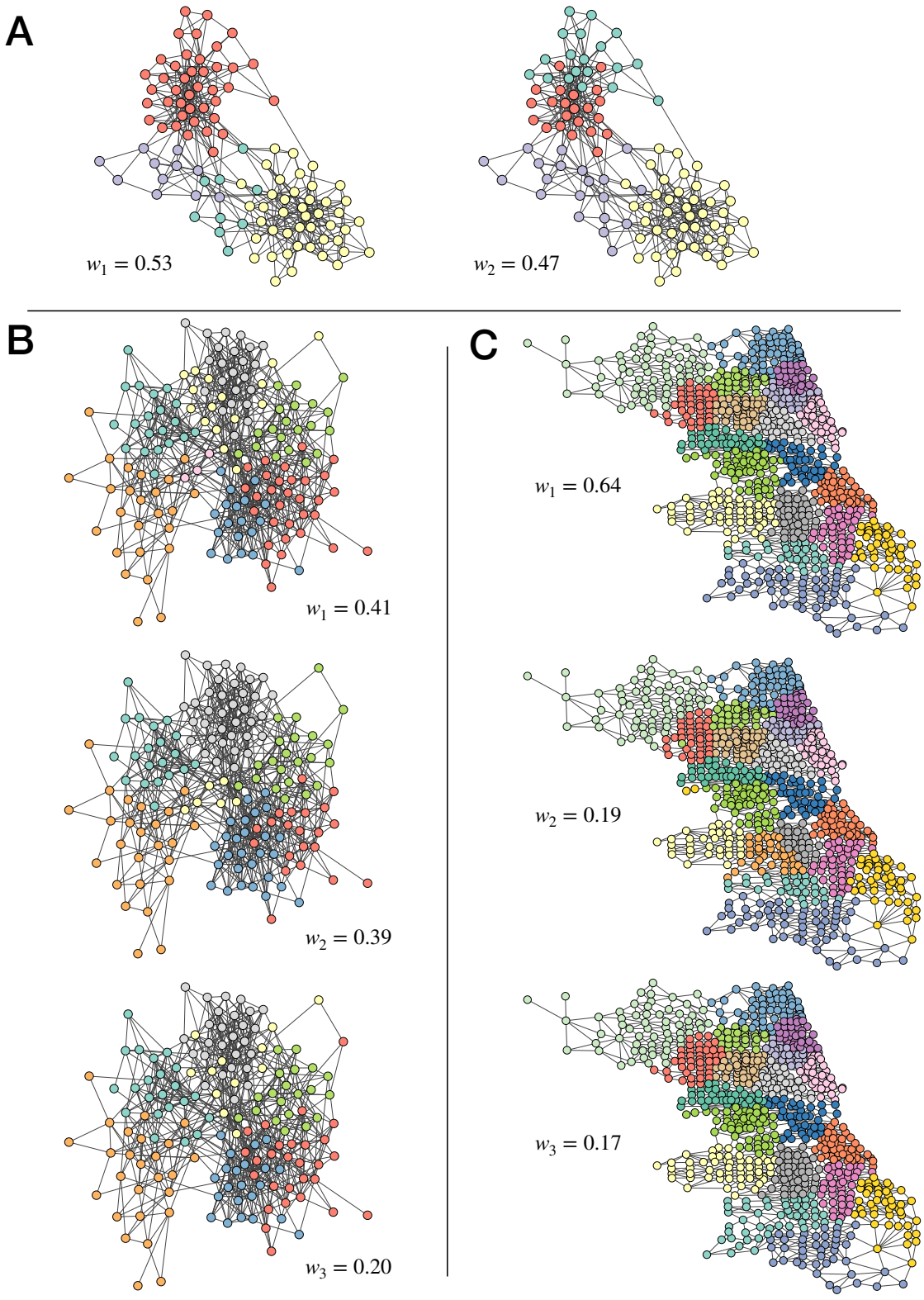

Fig. 5.3. Representative modes and their corresponding weights for three real-world example networks, identified by minimizing Eq. 5.6 with $\lambda = 1$ for $10,000$ community partition samples. (A) Network of political book co-purchases [59]. (B) High school friendship network [271, 272]. (C) Network of adjacent census tracts in the city of Chicago [273].

friendships among US high school students drawn from the National Longitudinal Study of Adolescent to Adult Health (the "Add Health" study) [271, 272]. The particular network we examine here is network number 5 from the study with 157 students. (Two nodes with degree zero were removed from the network before running the analysis.) As the figure shows, the method in this case finds three modes, each composed of half a dozen core communities of highly connected nodes whose boundaries shift somewhat from one mode to another, as well as a set of centrally located nodes (pale pink and yellow in the figure) that seem to move between communities in different modes. The movement of nodes from one community to another may be a sign of different roles played by core and peripheral members of social circles, or of students with a broad range of friendships.

In Fig. 5.3C, we show a third type of network, the geographic network of census tracts in the city of Chicago used for community detection in a different context in Sec. 3.3.3 [273]. (In contrast with the results in Sec. 3.3.3, we do not use weights in the community detection procedure used here, for consistency with the rest of the examples.) We recall that in this network the nodes represent the census tracts and two nodes are joined by an edge if the two corresponding tracts share a border, and community detection applied to this network tends to find contiguous local neighborhoods. Our algorithm finds three modes that differ primarily in the communities on the southwest side of the city where the density of census tracts is lower (though it is unclear whether this is the driving factor in the variation of community structure).

5.4. Conclusion

In this chapter we have presented a method for summarizing the complex output of community detection algorithms that identifies a small number of archetypal network partitions that are broadly representative of high-scoring partitions in general. The method is based on fundamental information theoretic principles, employing a clustering objective function based on the description length required to transmit a set of partitions using a specific multi-step encoding that we describe. We have developed an efficient algorithm to minimize this objective and we give examples of applications to both synthetic and real-world networks that exhibit nontrivial multimodal community structure.

Chapter 6.

Conclusion

My goal with the work in this thesis is to bridge current gaps between the theory and practice of network science by developing principled methods for incorporating metadata into the analysis of networks and formulating efficient algorithms to extract information from network data through statistical inference. In terms of theoretical contributions, my hope is that this work can provide new insights about networked systems that are obscured by existing measures, and that the ideas I have presented are sufficiently fundamental and intuitive to inspire further refinement and adaptation to new problems. On the more practical side, I aim to expand the mathematical and computational toolkit for researchers using networks in their work by providing measures that quantify the interplay of network topology and metadata, as well as algorithms to make otherwise intractable inference tasks more accessible (both computationally and intuitively).

In Chapter 2, I presented new measures for assessing balance in signed networks, showing that real networks are indeed structurally balanced and that we can use this information to effectively predict missing data. Many extensions and generalizations

of the work presented here would be possible. Good data on signed networks are currently relatively scarce, but it would be interesting to see how the results generalize when similar calculations are performed on other networks, particularly social networks. As discussed in Section 2.3.1, many data sets are more naturally represented as weighted and/or directed signed networks, and so extending the measures proposed here to these classes of networks would provide a more flexible framework for analysis of a wide variety of data. One could also employ balance metrics to perform anomaly detection in networks, looking for edges that participate in a large number of imbalanced loops. A further interesting question is how to determine the optimal value of the parameter α, which controls the amount by which longer loops are discounted in the calculations. In this work I simply chose a value that seems reasonable, noting that the results are not strongly dependent on the choice, but it would be an improvement if one were able to find a first-principles method of fixing the value of α. Finally, adapting the proposed measures to count only simple cycles rather than all closed walks could potentially improve their performance for less penalizing discount parameters α.

In Chapter 3, I discussed a framework for measuring the variation in distributional metadata across networks, demonstrating its potential to identify a variety of interesting patterns in spatial socioeconomic data. There are numerous improvements that can be made to this methodology in future work, particularly to increase its effectiveness in practical applications. Firstly, important limitations arise from the quality and resolution of census data, which I do not attempt to address. In particular, the coarse binning of interval distributional datasets (in this case, income and housing) can result in poor estimation of entropy and other uncertainty measures, as long tails are not accounted for and these tails may account for a large portion of the variabil-

ity in the distributions [274]. One improvement to the methodology to obtain more accurate results would thus be to estimate these full distributions based on the predefined bins and other summary statistics such as the mean, median, and Gini coefficient [154, 275], then apply the new measures using approximations of differential entropy. Additionally, some census data have large margins of error due to various statistical sampling issues [276, 277], and so correcting for this noise in the analyses would also improve the efficacy of these techniques. More generally, this framework is applicable to any network with node metadata that take the form of a distribution, and so it would be interesting to see the techniques presented in this chapter applied to other systems. One example is scientific collaboration networks, where each researcher has a distribution of disciplines in which they have previously published, which can be used to assess mixing preferences among co-authors [278].

In Chapter 4, I derived a new message passing algorithm for the solution of probabilistic models on networks containing short loops, which gives good results in both real and synthetic example applications, in contrast to standard message passing which fails badly on such networks. There are many ways in which the methods and results of this study could be extended. I studied only one application in detail, the Ising model, but the formalism presented is a general one that could be applied to many other models, including those with more general factor graph representations. In principle, any model with sparse pairwise interactions (i.e., interactions whose number scales sub-quadratically with the number of variables) could be studied using these methods. For example, there is a large class of generative models of networks in which edges appear with probabilities that depend on the properties of the adjacent nodes. Examples include the Chung-Lu model [279] and the stochastic block model and its variants [67, 77]. If we assume an observed network to be drawn from such

a model then we can use statistical inference to estimate the values of hidden node attributes that influence edge probability, such as community membership. The proposed message passing methods could be applied to such inference calculations and could in principle give more accurate results in the common case where the observed network contains many short loops. Another potential application in the realm of statistical inference is the inverse Ising model, the problem of inferring the parameters of an Ising or Ising-like model from an observed sequence of spin states, which has numerous applications including the reconstruction of neural pathways [280], the inference of protein structure [281], and correlations within financial markets [282]. It can be shown that the one- and two-point correlation functions of the observed spins are sufficient statistics to reliably estimate coupling and external field parameters [283] and the described method could be used to compute these statistics on loopy networks to greater accuracy than with traditional message passing and faster than standard Monte Carlo simulation. Other potential applications, further afield from traditional statistical physics, include the solution of constraint satisfaction problems, coding theory, and combinatorial optimization.

In Chapter 5, I concluded this thesis by presenting a simple, efficient summarization procedure based on information theoretic arguments which can extract a set of representative community divisions that capture the plausible partitions of a network. One can envisage many potential applications of this approach. As mentioned in Sec. 5.3.2, the representative community partitions for a social network could highlight distinct roles or reveal information about the diversity of a node's social circle. In networks with additional node metadata one could investigate how individual attributes are associated with the representative partitions. Multimodal community structure may also be of interest in spatial networks, for instance for assessing competing partitions,

as in mesh segmentation in engineering and computer graphics [284]. More generally, in the same way that any measurement can be supplemented with an error estimate, any community structure analysis could be supplemented with an analysis of competing partitions to help understand whether the optimal division is representative of the structure of the network as a whole. Additionally, the techniques presented in this study could be extended in a number of ways. The concept underlying the described algorithm is applicable to any set of partitions—not just community divisions of a network but partitions of any set of objects or data items—so it could be applied in any situation where there are multiple competing ways to cluster objects. All that is needed is an appropriate measure of the information required to encode representative objects and their corresponding clusters. One potential application within network science could be to the identification of representative networks within a set sampled from some generative model, such as an exponential random graph model [285].

Common to all the work contained in this thesis is the integration of knowledge from multiple disciplines. The notion of balanced triads from psychology can be extended to more general interaction structures through graph theory. Segregation, a fundamentally sociological and political phenomenon, can be unveiled with information theoretic comparison of demographic variables. The behavior of magnetic materials can be better understood by considering the loop structure of graphs underlying their spin interactions. And uncertainty in the modular structure of social systems can be quantified by identifying a maximally efficient information encoding for transmission to a receiver. It is critical to merge multiple disciplinary perspectives in order to bridge the gap between theoretical and application-oriented network science, as it is an inherently interdisciplinary field. I aim to spark interest in a broad audience with the questions I tackle, and I hope the work I have presented here will

form a foundation for new and exciting collaborations.

Appendix A.

Community Inference Methods

In this appendix we discuss inference methods to handle the intractability of the posterior distribution $P(\mathbf{g}, \boldsymbol{\theta}|G)$ in Eq. 1.15. These methods include Monte Carlo sampling, variational Bayesian inference, profile likelihood estimation, and the expectation maximization algorithm. We then give an example of Bayesian community inference for the stochastic block model using expectation maximization with belief propagation.

A.1. Algorithmic techniques

Perhaps the simplest solution to the intractability of $P(\mathbf{g}, \boldsymbol{\theta}|G)$, and the one most commonly used in practice, is just to ignore the evidence $P(G)$ altogether. If we can draw samples from $P(\mathbf{g}, \boldsymbol{\theta}|G)$, we can construct estimates of the model variables $\mathbf{g}, \boldsymbol{\theta}$ and compute expectation values over this posterior distribution. This sampling only requires us to know the *ratio* of the posterior probability of two configurations $\mathbf{g}, \boldsymbol{\theta}$ and $\mathbf{g}', \boldsymbol{\theta}'$, which crucially does not depend on the denominator $P(G)$. Monte Carlo

methods are used to perform this sampling, and for a given inference task there are numerous ways one can take samples that are informative of the posterior distribution. (See [286] for details on Monte Carlo sampling and its variants in the context of statistical physics systems.) For a judicious choice of prior (typically one uses a *conjugate prior* with respect to relevant terms in the likelihood [78]), it may be possible to average over some or all of the parameters $\boldsymbol{\theta}$, in which case we can reduce the dimensionality of the space we need to sample from, at the cost of not obtaining estimates for the parameters that have been integrated out.

Alternatively, we can circumvent the integral in Eq. 1.15 by using a simpler distribution $\tilde{P}(\mathbf{g}, \boldsymbol{\theta})$ as an approximation for $P(\mathbf{g}, \boldsymbol{\theta}|G)$. The form of this approximating distribution is typically restricted to a simple family of distributions (e.g. a product of independent probability distributions), and our goal is to identify the specific distribution $\tilde{P}(\mathbf{g}, \boldsymbol{\theta})$ within this family that minimizes the *Kullback Leibler (KL) divergence*

$$D_{KL}(\tilde{P}||P) = \sum_{\mathbf{g},\boldsymbol{\theta}} \tilde{P}(\mathbf{g}, \boldsymbol{\theta}) \log \frac{\tilde{P}(\mathbf{g}, \boldsymbol{\theta})}{P(\mathbf{g}, \boldsymbol{\theta}|G)}, \tag{A.1}$$

where we've used as shorthand a summation sign to indicate sums/integrals over discrete/continuous variables respectively. After some manipulation, one can show that this minimization is equivalent to maximizing the *evidence lower bound* (also known as the negative *variational free energy*)

$$\text{ELBO}(G) = -\sum_{\mathbf{g},\boldsymbol{\theta}} \tilde{P}(\mathbf{g}, \boldsymbol{\theta}) \log \tilde{P}(\mathbf{g}, \boldsymbol{\theta}) + \sum_{\mathbf{g},\boldsymbol{\theta}} \tilde{P}(\mathbf{g}, \boldsymbol{\theta}) \log[P(G|\mathbf{g}, \boldsymbol{\theta}) P(\mathbf{g}, \boldsymbol{\theta})]. \tag{A.2}$$

Maximizing Eq. A.2 results in a set of self-consistent equations for the parameters of the distribution $\tilde{P}$ that can be solved using numerical iteration. Due to its usage of

the calculus of variations to derive optimization conditions, this technique is called *variational Bayesian inference* [78].

Fortunately, in many cases we may only be interested in point estimates (single values) of the parameters $\boldsymbol{\theta}$, in which case we have multiple additional methods at our disposal. The simplest is usually to compute the *maximum a posteriori (MAP)* estimator

$$\hat{\boldsymbol{\theta}}(\mathbf{g}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \{P(\mathbf{g}, \boldsymbol{\theta}|G)\} \propto \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \{P(G|\mathbf{g}, \boldsymbol{\theta})P(\mathbf{g}, \boldsymbol{\theta})\} \tag{A.3}$$

as a function of the group assignments $\mathbf{g}$, then feed this estimator back into the posterior distribution to obtain the *profile likelihood*

$$P(\mathbf{g}, \hat{\boldsymbol{\theta}}(\mathbf{g})|G) \propto P(G|\mathbf{g}, \hat{\boldsymbol{\theta}}(\mathbf{g}))P(\mathbf{g}, \hat{\boldsymbol{\theta}}(\mathbf{g})), \tag{A.4}$$

which is then only a function of the group assignments. This distribution can be sampled using Monte Carlo methods, or (to obtain point estimates of $\mathbf{g}$) optimized using a variety of stochastic optimization methods.

An alternative technique for point estimation of $\boldsymbol{\theta}$, which resembles variational Bayesian inference, is called the *expectation-maximization* (EM) algorithm [287]. In the EM algorithm, we seek the *marginal MAP* estimator $\hat{\boldsymbol{\theta}}$, given by

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left\{ \sum_{\mathbf{g}} P(\mathbf{g}, \boldsymbol{\theta}|G) \right\} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left\{ \log \sum_{\mathbf{g}} P(\mathbf{g}, \boldsymbol{\theta}|G) \right\}, \tag{A.5}$$

where we have taken the logarithm of the marginal posterior as the objective function. This transformation does not change the optimization problem due to the monotone increasing nature of the logarithm, and it will facilitate the derivation of the equations we need for optimization. For the same reason as the model evidence, we cannot

evaluate the term inside the brackets analytically or numerically. However, as in the case of variational Bayesian inference, we can maximize a lower bound on this sum to get our desired result. Using Jensen's inequality, we can write

$$\log \sum_{\mathbf{g}} P(\mathbf{g}, \boldsymbol{\theta}|G) \geq \sum_{\mathbf{g}} q(\mathbf{g}) \log[P(\mathbf{g}, \boldsymbol{\theta}|G)/q(\mathbf{g})], \tag{A.6}$$

where $q(\mathbf{g})$ is any properly normalized probability distribution over partitions $\mathbf{g}$. By inspection, we can see that the inequality is satisfied when

$$q(\mathbf{g}) = \frac{P(\mathbf{g}, \boldsymbol{\theta}|G)}{\sum_{\mathbf{g}} P(\mathbf{g}, \boldsymbol{\theta}|G)}. \tag{A.7}$$

With this choice of $q(\mathbf{g})$, the parameters $\boldsymbol{\theta}$ that maximize the right hand side of Eq. A.6 are the marginal MAP estimators $\hat{\boldsymbol{\theta}}$ we are looking for. These can be computed using the equations

$$\sum_{\mathbf{g}} q(\mathbf{g}) \nabla_{\boldsymbol{\theta}} \log P(\mathbf{g}, \boldsymbol{\theta}|G)|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} = 0, \tag{A.8}$$

where $\nabla_{\boldsymbol{\theta}}$ is the gradient operator. Since the left hand side of Eq. A.8 is just an expectation value over the distribution $q(\mathbf{g})$, it can be evaluated using Monte Carlo sampling. The EM algorithm in this case consists of initializing a guess for $\hat{\boldsymbol{\theta}}$, then (until convergence) alternating the steps of (1) sampling from $q(\mathbf{g})$ using Eq. A.7 and (2) updating the estimates of $\hat{\boldsymbol{\theta}}$ using A.8.

Alternatively, if the expectation in Eq A.8 can be written in terms of simple expectations over $q(\mathbf{g})$ (such as one- or two-point marginals), we can sometimes skip sampling altogether and instead use numerical iteration to estimate these posterior expectations, and thus the parameter updates. (We will show this in the upcoming

example.) This approach is typically much faster in practice, but does not always give good answers if we need to approximate the posterior statistics, which is typically the case for networks that have loops. In any case, the EM algorithm turns the single intractable maximization in Eq. A.5 into two tractable, interlaced maximization problems, which in the end gives us the marginal MAP estimates we want. In fact, we actually get more than just these point estimates for the model parameters, we get the whole posterior distribution of $\mathbf{g}$, conditioned on these estimates. We can see this by noticing that $q(\mathbf{g})$ in Eq. A.7 is none other than $P(\mathbf{g}|G, \boldsymbol{\theta})$, and so once our EM algorithm has reached convergence we have computed this distribution (or at least some of its useful expectation values) as well.

A.2. Belief propagation and inference with the SBM

Now that we've established the formalism one uses to perform Bayesian inference for community detection in networks, we can demonstrate how this works in practice using the stochastic block model as an example. We will use an expectation maximization algorithm with a numerical approximation technique known as *belief propagation* to solve for the marginal MAP estimates of the model parameters.

As described in Sec. 1.3.1, in the SBM with K groups, an undirected, unweighted edge is placed independently at random between each pair of nodes i and j with probability ω_{g_i,g_j}, where g_i is the group assignment of i and $\boldsymbol{\omega}$ is the $K \times K$ mixing matrix of edge probabilities between groups. In this case, $\boldsymbol{\theta} = \boldsymbol{\omega}$ and the probability of observing a graph G (with adjacency matrix $\mathbf{A}$), given the model variables $\mathbf{g}$ and

$\boldsymbol{\theta}$, is

$$P(G|\mathbf{g}, \boldsymbol{\omega}) = \prod_{i<j} \omega_{g_i g_j}^{A_{ij}} (1 - \omega_{g_i g_j})^{1 - A_{ij}}. \tag{A.9}$$

Now, if we put uniform priors on both the group assignments $\mathbf{g}$ and the mixing matrix $\boldsymbol{\omega}$ (to assume as little information as possible), we can write the model posterior as

$$P(\mathbf{g}, \boldsymbol{\omega}|G) = \frac{P(G|\mathbf{g}, \boldsymbol{\omega}) P(\mathbf{g}) P(\boldsymbol{\omega})}{P(G)} \propto P(G|\mathbf{g}, \boldsymbol{\omega}), \tag{A.10}$$

where we've ignored the evidence $P(G)$ as it will not play a role in our EM algorithm.

We can now substitute Eq. A.9 into Eq. A.8 to get the parameter update equation

$$\begin{aligned}
&\sum_{\mathbf{g}} q(\mathbf{g}) \frac{\partial}{\partial \omega_{rs}} \left[\sum_{i<j} (A_{ij} \log \omega_{g_i g_j} + (1 - A_{ij}) \log(1 - \omega_{g_i g_j})) \right] \Big|_{\omega_{rs} = \hat{\omega}_{rs}} \\
&\quad = \sum_{\mathbf{g}} q(\mathbf{g}) \left[\sum_{i<j} (A_{ij} \delta_{g_i, r} \delta_{g_j, s} / \hat{\omega}_{rs} - (1 - A_{ij}) \delta_{g_i, r} \delta_{g_j, s} / (1 - \hat{\omega}_{rs})) \right] \\
&\quad = \sum_{i<j} q_{ij}(r, s) [A_{ij} / \hat{\omega}_{rs} - (1 - A_{ij}) / (1 - \hat{\omega}_{rs})] \\
&\quad = 0 \\
&\Rightarrow \hat{\omega}_{rs} = \frac{\sum_{(i,j) \in E} q_{ij}(r, s)}{\sum_{i<j} q_{ij}(r, s)} = \frac{2 \sum_{(i,j) \in E} q_{ij}(r, s)}{\sum_{i,j} q_{ij}(r, s)}
\end{aligned} \tag{A.11}$$

where E is the edge set of G (which we've assumed has no self-edges) and $q_{ij}(r, s)$ is the posterior probability under $q(\mathbf{g})$ that i and j are simultaneously in groups r and s respectively.

This is the exact parameter update equation for our EM algorithm, but it is a bit computationally costly to evaluate, since for the denominator we have to compute

$q_{ij}(r,s)$ for all node pairs i,j. However, if we assume that for large networks the distribution of group sizes $\{n_r\}_{r=1}^K$ is tightly peaked, we can make the approximation

$$\begin{aligned}\sum_{i,j} q_{ij}(r,s) &= \sum_{\mathbf{g}} q(\mathbf{g}) \left[\sum_i \delta_{g_i,r} \sum_j \delta_{g_i,r}\right] = \sum_{\mathbf{g}} q(\mathbf{g})[n_r n_s] = \langle n_r n_s\rangle \\ &\approx \langle n_r\rangle\langle n_s\rangle = \sum_i q_i(r) \sum_j q_j(s), \end{aligned} \tag{A.12}$$

where $q_i(r)$ is the posterior probability under $q(\mathbf{g})$ that i is in group r. Our update equation now reads

$$\widehat{\omega}_{rs} = \frac{2\sum\limits_{(i,j)\in E} q_{ij}(r,s)}{\sum_i q_i(r)\sum_j q_j(s)}, \tag{A.13}$$

which can be evaluated in $\mathrm{O}(m+n)$ time.

The one- and two-point marginals $q_i(r)$ and $q_{ij}(r,s)$ in Eq. A.13 can be evaluated by Monte Carlo sampling $q(\mathbf{g})$, but for large networks this can be very slow. We can do better by computing them in an iterative manner using a set of self-consistent equations called the *belief propagation* equations [288]. These equations can be derived by assuming that the network is a tree—or in other words, has no simple cycles. (This is sometimes not a good assumption for networks with high clustering but we will ignore this issue for now; see Ch. 4 for more details on an improved solution.) Going back to the definition of $q_i(r)$, we have that

$$q_i(r) \propto \sum_{\mathbf{g}\,:\,g_i=r} P(\mathbf{g}|G,\boldsymbol{\omega}) = \sum_{\mathbf{g}\,:\,g_i=r} e^{\sum_{(i,j)\in E}\log\omega_{g_i,g_j}+\sum_{(i,j)\in\tilde{E}}\log(1-\omega_{g_i,g_j})}, \tag{A.14}$$

where the set $\mathbf{g} : g_i = r$ is the set of all possible partitions in which $g_i = r$, and $\tilde{E}$ is

the set of all node pairs not connected by an edge. For the moment, we will ignore the non-edge terms $\log(1-\omega_{g_i,g_j})$, as they will be dealt with later on with a mean-field approximation. Further manipulation of $q_i(r)$ then yields

$$\begin{aligned}\tilde{q}_i(r) &\propto \sum_{\mathbf{g}\,:\,g_i=r}\prod_{j\in\partial_i} e^{\log\omega_{g_i,g_j}}\prod_{k\in\partial_{j\setminus i}} e^{\log\omega_{g_j,g_k}}\prod_{l\in\partial_{k\setminus j}} e^{\log\omega_{g_k,g_l}}\ \cdots \\ &= \prod_{j\in\partial_i}\sum_{s=1}^{K} e^{\log\omega_{r,s}}\prod_{k\in\partial_{j\setminus i}}\sum_{t=1}^{K} e^{\log\omega_{s,t}}\prod_{l\in\partial_{k\setminus j}}\sum_{u=1}^{K} e^{\log\omega_{t,u}}\ \cdots, \end{aligned} \tag{A.15}$$

where ∂_i is the set of neighbors of i and $\partial_{j\setminus i}$ is the set of j's neighbors that are not neighbors with i—in the tree approximation we've made, this is the same as j's neighbors other than i, since i and j do not share any common neighbors. We've also notated the marginal probability as $\tilde{q}_i(r)$ to emphasize that this expression is not accounting for the effect of non-edges. Now, we can see by inspection that Eq. A.15 can be written in the following recursive form

$$\tilde{q}_i(r) \propto \prod_{j\in\partial_i}\sum_{s}\omega_{rs}q_{i\leftarrow j}(s), \tag{A.16}$$

where

$$\tilde{q}_{i\leftarrow j}(s) \propto \prod_{k\in\partial_{j\setminus i}}\sum_{t}\omega_{st}q_{j\leftarrow k}(t) \tag{A.17}$$

is the marginal posterior probability that j is in group s if i is removed from the network. This is also known as a "message", which is passed from j to i.

Now, if $\omega_{rs}\sim O(1/n)$, as is the case for a sparse network with $m\sim O(n)$, then up to sub-leading terms in n, i sends the same message to all of its non-neighbors j,

which is simply its marginal $q_i(r)$ [288]. To account for all interactions contributing to node i's marginal probability, we can thus tack on to Eq. A.16 the product of these independent messages coming from all of i's non-neighbors. Making the analogous transformation for the equations for the messages $q_{i\leftarrow j}$, our final belief propagation equations are given by

$$q_i(r) = \frac{1}{Z_i} \prod_{j \notin \partial_i} \left[1 - \sum_s \omega_{rs} q_j(s) \right] \prod_{j \in \partial_i} \sum_s \omega_{rs} q_{i \leftarrow j}(s) \tag{A.18}$$

and

$$q_{i\leftarrow j}(s) = \frac{1}{Z_{i\leftarrow j}} \prod_{k \notin \partial_{j \setminus i}} \left[1 - \sum_t \omega_{st} q_k(t) \right] \prod_{k \in \partial_{j \setminus i}} \sum_t \omega_{st} q_{j \leftarrow k}(t), \tag{A.19}$$

where

$$Z_i = \sum_r \prod_{j \notin \partial_i} \left[1 - \sum_s \omega_{rs} q_j(s) \right] \prod_{j \in \partial_i} \sum_s \omega_{rs} q_{i \leftarrow j}(s) \tag{A.20}$$

and

$$Z_{i\leftarrow j} = \sum_s \prod_{k \notin \partial_{j \setminus i}} \left[1 - \sum_t \omega_{st} q_k(t) \right] \prod_{k \in \partial_{j \setminus i}} \sum_t \omega_{st} q_{j \leftarrow k}(t), \tag{A.21}$$

are computed upon each update of the messages and marginals to ensure that these are properly normalized. Our update for $q_i(r)$ now consists of iterating Eq.s A.18 and A.19 until convergence.

To update the parameters $\widehat{\omega}_{rs}$ in Eq. A.13 for our EM algorithm, we also need the two-point marginals $q_{ij}(r, s)$ for the edges $(i, j) \in E$, which can be computed after

the convergence of the messages $q_{i \leftarrow j}$. Since we are assuming G is a tree, removal of the edge (i, j) will split G into two sub-trees, one rooted at i and the other at j. The probability $q_{ij}(r, s)$ that connected nodes i and j are in groups r and s respectively is then proportional to the probability that i is in group r with j removed from the network, times the probability that j is in group s with i removed from the network, times the probability that i and j are connected by an edge given that they are in groups r and s. Mathematically, we have

$$q_{ij}(r, s) = \frac{q_{j \leftarrow i}(r) q_{i \leftarrow j}(s) \omega_{rs}}{Z_{ij}}, \tag{A.22}$$

where

$$Z_{ij} = \sum_{r,s} q_{j \leftarrow i}(r) q_{i \leftarrow j}(s) \omega_{rs} \tag{A.23}$$

normalizes the two-point marginal. The complete EM algorithm for inferring the marginal MAP estimators in the SBM involves alternating the parameter update in Eq. A.13 with the belief propagation updates in Eq.s A.18 and A.22, until convergence. At the end of this process, we then have both $\hat{\boldsymbol{\omega}}$ as well as the posterior marginals q_i and q_{ij}.

Appendix B.

Supplementary Material for Chapter 4

In this appendix we provide detailed derivations for multiple results in Chapter 4, including the specific heat, local Monte Carlo sampling algorithm, message passing Jacobian, and entropy.

B.1. Calculation of the heat capacity using message passing

The heat capacity, which is given by

$$C = \frac{dU}{dT} = -\beta^2 \frac{dU}{d\beta}, \tag{B.1}$$

can be calculated from the expression for the internal energy

$$U(\beta) = \frac{1}{2} \sum_{i \in V} \frac{1}{Z_i(\beta)} \sum_{\mathbf{s}_{N_i}} H_{\partial_i}(\mathbf{s}_{\partial_i})\, e^{-\beta H_{N_i}(\mathbf{s}_{N_i})} \prod_{j \in N_i \setminus i} q_{i \leftarrow j}(s_j, \beta),$$

where instead of incorporating the β dependence into the Hamiltonian as in the chapter, we now temporarily it explicitly for clarity of demonstration. In this expression, N_i denotes the neighborhood of node i as in the main text, ∂_i denotes the node i and its immediately adjacent edges and nodes, and $H_{N_i}(\mathbf{s}_{N_i})$ and $H_{\partial_i}(\mathbf{s}_{\partial_i})$ represent the terms in the Hamiltonian for these subgraphs:

$$H_{N_i}(\mathbf{s}_{N_i}) = -f_i(s_i|\theta_i) - \sum_{(j,k)\in N_i} g_{jk}(s_j, s_k|\omega_{jk}) \tag{B.2}$$

and

$$H_{\partial_i}(\mathbf{s}_{\partial_i}) = -2f_i(s_i|\theta_i) - \sum_{(i,j)\in \partial_i} g_{ij}(s_i, s_j|\omega_{ij}), \tag{B.3}$$

with the β dependence omitted from the definition of the functions. With the β dependence written in this way the message passing equations take the form

$$q_i(x,\beta) = \frac{1}{Z_i(\beta)} \sum_{\mathbf{s}_{N_i\setminus i}} \delta_{s_i,x} e^{-\beta H_{N_i}(\mathbf{s}_{N_i})} \prod_{j\in N_i\setminus i} q_{i\leftarrow j}(s_j,\beta), \tag{B.4}$$

and

$$q_{i\leftarrow j}(y,\beta) = \frac{1}{Z_{i\leftarrow j}(\beta)} \sum_{\mathbf{s}_{N_{j\setminus i}}} \delta_{s_j,y} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})} \prod_{k\in N_{j\setminus i}\setminus j} q_{j\leftarrow k}(s_k,\beta), \tag{B.5}$$

with

$$Z_i(\beta) = \sum_{\mathbf{s}_{N_i}} e^{-\beta H_{N_i}(\mathbf{s}_{N_i})} \prod_{j \in N_i \setminus i} q_{i \leftarrow j}(s_j, \beta), \tag{B.6}$$

$$Z_{i \leftarrow j}(\beta) = \sum_{\mathbf{s}_{N_{j \setminus i}}} e^{-\beta H_{N_{j \setminus i}}(\mathbf{s}_{N_{j \setminus i}})} \prod_{k \in N_{j \setminus i} \setminus j} q_{j \leftarrow k}(s_k, \beta). \tag{B.7}$$

Differentiating B.5 with respect to β and defining the quantity

$$\eta_{i \leftarrow j}(y) = \frac{dq_{i \leftarrow j}(y, \beta)}{d\beta}, \tag{B.8}$$

we get

$$\begin{aligned} \eta_{i \leftarrow j}(y) = \frac{1}{Z_{i \leftarrow j}} \sum_{\mathbf{s}_{N_{j \setminus i}}} e^{-\beta H_{N_{j \setminus i}}(\mathbf{s}_{N_{j \setminus i}})} \prod_{k \in N_{j \setminus i} \setminus j} q_{j \leftarrow k}(s_k) \Bigg(\left[q_{i \leftarrow j}(y) - \delta_{s_j, y} \right] H_{N_{j \setminus i}}(\mathbf{s}_{N_{j \setminus i}}) \\ + \left[\delta_{s_j, y} - q_{i \leftarrow j}(y) \right] \sum_{k \in N_{j \setminus i} \setminus j} \frac{\eta_{j \leftarrow k}(s_k)}{q_{j \leftarrow k}(s_k)} \Bigg), \end{aligned} \tag{B.9}$$

which can be regarded as a new message passing equation for the derivative $\eta_{i \leftarrow j}(y)$. To apply it, we first solve for the $q_{i \leftarrow j}(y)$ in the usual fashion then fix their values and iterate (B.9) from a suitable initial condition until convergence.

For large neighborhoods, where the sums over spins states cannot be performed exhaustively, the local Monte Carlo procedure described in the main text carries over naturally. We define

$$\langle A \rangle_{N_{j \setminus i}} = \sum_{\mathbf{s}_{N_{j \setminus i}}} A(\mathbf{s}_{N_{j \setminus i}}) \frac{e^{-\beta H_{N_{j \setminus i}}(\mathbf{s}_{N_{j \setminus i}})} \prod_{k \in N_{j \setminus i} \setminus j} q_{j \leftarrow k}(s_k)}{Z_{i \leftarrow j}} \tag{B.10}$$

and then rewrite Eq. (B.9) as an average

$$\eta_{i\leftarrow j}(y) = \left\langle \left[q_{i\leftarrow j}(y) - \delta_{s_j,y}\right] H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}}) + \left[\delta_{s_j,y} - q_{i\leftarrow j}(y)\right] \sum_{k\in N_{j\setminus i}\setminus j} \frac{\eta_{j\leftarrow k}(s_k)}{q_{j\leftarrow k}(s_k)} \right\rangle_{N_{j\setminus i}}, \tag{B.11}$$

which can be evaluated using Monte Carlo sampling as previously.

We can also differentiate $Z_i(\beta)$, Eq. (B.6), which yields

$$\begin{aligned} \frac{1}{Z_i(\beta)}\frac{dZ_i(\beta)}{d\beta} = \frac{1}{Z_i}\sum_{\mathbf{s}_{N_i}} e^{-\beta H_{N_i}(\mathbf{s}_{N_i})} \prod_{j\in N_i\setminus i} q_{i\leftarrow j}(s_j) \\ \times \left[\sum_{j\in N_i\setminus i} \frac{1}{q_{i\leftarrow j}(s_j)}\frac{dq_{i\leftarrow j}(s_j,\beta)}{d\beta} - H_{N_i}(\mathbf{s}_{N_i})\right], \end{aligned} \tag{B.12}$$

which can again be written as an average

$$\frac{1}{Z_i(\beta)}\frac{dZ_i(\beta)}{d\beta} = \left\langle \sum_{j\in N_i\setminus i} \frac{\eta_{i\leftarrow j}(s_j)}{q_{i\leftarrow j}(s_j)} - H_{N_i}(\mathbf{s}_{N_i}) \right\rangle_{N_i}, \tag{B.13}$$

where we have used a shorthand analogous to that of Eq. (B.10):

$$\langle A\rangle_{N_i} = \sum_{\mathbf{s}_{N_i}} A(\mathbf{s}_{N_i}) \frac{e^{-\beta H_{N_i}(\mathbf{s}_{N_i})} \prod_{j\in N_i\setminus i} q_{i\leftarrow j}(s_j)}{Z_i}. \tag{B.14}$$

Differentiating Eq. (B.2) and substituting from Eqs. (B.9) and (B.13) we now find,

after some manipulation, that

$$\frac{dU}{d\beta} = \frac{1}{2}\sum_{i\in V}\left[\left\langle H_{\partial_i}(\mathbf{s}_{\partial_i})\right\rangle_{N_i}\left\langle H_{N_i}(\mathbf{s}_{N_i})\right\rangle_{N_i} - \left\langle H_{\partial_i}(\mathbf{s}_{\partial_i})H_{N_i}(\mathbf{s}_{N_i})\right\rangle_{N_i}\right] + \frac{1}{2}\sum_{i\in V}\left[\left\langle H_{\partial_i}(\mathbf{s}_{\partial_i})\sum_{j\in N_i\setminus i}\frac{\eta_{i\leftarrow j}(s_j)}{q_{i\leftarrow j}(s_j)}\right\rangle_{N_i} - \left\langle H_{\partial_i}(\mathbf{s}_{\partial_i})\right\rangle_{N_i}\left\langle\sum_{j\in N_i\setminus i}\frac{\eta_{i\leftarrow j}(s_j)}{q_{i\leftarrow j}(s_j)}\right\rangle_{N_i}\right], \tag{B.15}$$

which can be substituted into Eq. (B.1) to calculate C.

B.2. Local Monte Carlo simulation for the Ising model

As discussed in the main text, when neighborhoods are too large to allow us to sum exhaustively over their states we can approximate the message passing equations by Monte Carlo sampling. Taking again the example of the Ising model, the message passing equations are

$$q_i = \frac{\sum_{\mathbf{s}_{N_i}}\delta_{s_i,+1}e^{-\beta H_{N_i}(\mathbf{s}_{N_i})}\prod_{j\in N_i\setminus i}q_{i\leftarrow j}(s_j)}{\sum_{\mathbf{s}_{N_i}}e^{-\beta H_{N_i}(\mathbf{s}_{N_i})}\prod_{j\in N_i\setminus i}q_{i\leftarrow j}(s_j)}, \tag{B.16}$$

$$q_{i\leftarrow j} = \frac{\sum_{\mathbf{s}_{N_{j\setminus i}}}\delta_{s_j,+1}e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}\prod_{k\in N_{j\setminus i}\setminus j}q_{j\leftarrow k}(s_k)}{\sum_{\mathbf{s}_{N_{j\setminus i}}}e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}\prod_{k\in N_{j\setminus i}\setminus j}q_{j\leftarrow k}(s_k)}, \tag{B.17}$$

where the messages in this case represent the probability of the corresponding spin being $+1$. If we divide top and bottom by $\sum_{\mathbf{s}_{N_i}}e^{-\beta H_{N_i}(\mathbf{s}_{N_i})}$ in the first equation and

by $\sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}$ in the second, we get

$$q_i = \frac{\sum_{\mathbf{s}_{N_i}} e^{-\beta H_{N_i}(\mathbf{s}_{N_i})}(\delta_{s_i,+1}\prod_{j\in N_i\setminus i} q_{i\leftarrow j}(s_j))\big/\sum_{\mathbf{s}_{N_i}} e^{-\beta H_{N_i}(\mathbf{s}_{N_i})}}{\sum_{\mathbf{s}_{N_i}} e^{-\beta H_{N_i}(\mathbf{s}_{N_i})}(\prod_{j\in N_i\setminus i} q_{i\leftarrow j}(s_j))\big/\sum_{\mathbf{s}_{N_i}} e^{-\beta H_{N_i}(\mathbf{s}_{N_i})}}, \tag{B.18}$$

$$q_{i\leftarrow j} = \frac{\sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}(\delta_{s_j,+1}\prod_{k\in N_{j\setminus i}\setminus j} q_{j\leftarrow k}(s_k))\big/\sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}}{\sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}(\prod_{k\in N_{j\setminus i}\setminus j} q_{j\leftarrow k}(s_k))\big/\sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}}. \tag{B.19}$$

Numerators and denominators now take the form of a Boltzmann average, but over the distributions defined by H_{N_i} and $H_{N_{j\setminus i}}$ alone, which we can think of as a "zero-field" ensemble that omits the effect of the "external field" imposed by the messages. Defining the useful shorthand

$$\langle A\rangle_{0,N_i} = \frac{\sum_{\mathbf{s}_{N_i}} e^{-\beta H_{N_i}(\mathbf{s}_{N_i})} A(\mathbf{s}_{N_i})}{\sum_{\mathbf{s}_{N_i}} e^{-\beta H_{N_i}(\mathbf{s}_{N_i})}}, \tag{B.20}$$

$$\langle A\rangle_{0,N_{j\setminus i}} = \frac{\sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})} A(\mathbf{s}_{N_{j\setminus i}})}{\sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}}, \tag{B.21}$$

we can then write the message passing equations in the form

$$q_i = \frac{\big\langle \delta_{s_i,+1}\prod\limits_{j\in N_i\setminus i} q_{i\leftarrow j}(s_j)\big\rangle_{0,N_i}}{\big\langle \prod\limits_{j\in N_i\setminus i} q_{i\leftarrow j}(s_j)\big\rangle_{0,N_i}}, \tag{B.22}$$

$$q_{i\leftarrow j} = \frac{\big\langle \delta_{s_j,+1}\prod\limits_{k\in N_{j\setminus i}\setminus j} q_{j\leftarrow k}(s_k)\big\rangle_{0,N_{j\setminus i}}}{\big\langle \prod\limits_{k\in N_{j\setminus i}\setminus j} q_{j\leftarrow k}(s_k)\big\rangle_{0,N_{j\setminus i}}}, \tag{B.23}$$

where the "0" serves to remind us that the expectation is over the zero-field ensemble. Expressing the equations as zero-field expectations allows us to evaluate them using the Wolff algorithm, which is highly efficient in this context.

We can further speed up sampling by making use of the up-down symmetry of the zero-field ensemble, which effectively gives us two samples for every spin state. If we obtain a set of samples $\{\mathbf{s}_N\}$ by sampling from the zero-field ensemble, then because of symmetry $\{-\mathbf{s}_N\}$ are also correct samples that would have occurred with the same probability. Including these additional samples explicitly in the message passing equations gives

$$q_{i\leftarrow j} = \frac{\left\langle \delta_{s_j,+1} \prod\limits_{k\in N_{j\setminus i}\setminus j} q_{j\leftarrow k}(s_k) + \delta_{-s_j,+1} \prod\limits_{k\in N_{j\setminus i}\setminus j} (1-q_{j\leftarrow k}(s_k)) \right\rangle_{0,N_{j\setminus i}}}{\left\langle \prod\limits_{k\in N_{j\setminus i}\setminus j} q_{j\leftarrow k}(s_k) + \prod\limits_{k\in N_{j\setminus i}\setminus j} (1-q_{j\leftarrow k}(s_k)) \right\rangle_{0,N_{j\setminus i}}}, \tag{B.24}$$

and corresponding expressions can be derived for any expectation.

B.3. The Jacobian at the critical point

In the main text we used the leading eigenvalue of the Jacobian of the message passing iteration at the trivial fixed point to locate the position of the phase transition. Taking the Ising model as our example once again, the calculation is as follows.

The message passing equations can be rewritten as

$$q_{i\leftarrow j} = \frac{1}{Z_{i\leftarrow j}} \sum_{\mathbf{s}_{N_{j\setminus i}}} \tfrac{1}{2}(1+s_j)\, e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})} \prod_{k\in N_{j\setminus i}\setminus j} \left[\tfrac{1}{2}(1-s_k) + s_k q_{j\leftarrow k}\right],$$

where

$$Z_{i\leftarrow j} = \sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})} \prod_{k\in N_{j\setminus i}\setminus j} [\tfrac{1}{2}(1-s_k) + s_k q_{j\leftarrow k}]. \tag{B.25}$$

Considering the sum over spins as a local average again, the elements of the Jacobian are then given by

$$\frac{\partial q_{i\leftarrow j}}{\partial q_{\mu\leftarrow\nu}} = \mathbf{1}_{\{\mu=j,\nu\in N_{j\setminus i}\}}\left[\left\langle \frac{(1+s_j)s_\nu}{1-s_\nu+2s_\nu q_{\mu\leftarrow\nu}}\right\rangle_{N_{j\setminus i}} - \langle 1+s_j\rangle_{N_{j\setminus i}}\left\langle \frac{s_\nu}{1-s_\nu+2s_\nu q_{\mu\leftarrow\nu}}\right\rangle_{N_{j\setminus i}}\right], \tag{B.26}$$

where $\mathbf{1}_{\{\dots\}}$ is the indicator function and we have used the shorthand from Eq. (B.10) again. Now evaluating this expression at the trivial fixed point $q_{j\leftarrow k} = \frac{1}{2}$ for all j, k (which we write as simply $q = \frac{1}{2}$ for short), we get the Jacobian

$$J_{j\to i,\nu\to\mu} = \left.\frac{\partial q_{i\leftarrow j}}{\partial q_{\mu\leftarrow\nu}}\right|_{q=\frac{1}{2}} = \tilde{B}_{j\to i,\nu\to\mu} D_{j\to i,\nu\to\mu}, \tag{B.27}$$

where $\tilde{B}$ is a generalization of the non-backtracking matrix given by

$$\tilde{B}_{j\to i,\nu\to\mu} = \begin{cases} 1 & \text{if } \mu = j \text{ and } \nu \in N_{j\setminus i}, \\ 0 & \text{otherwise,} \end{cases} \tag{B.28}$$

and

$$D_{j\to i,\nu\to\mu} = \frac{\sum_{\mathbf{s}_{N_{j\setminus i}}} s_\mu s_\nu \, e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}}{\sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}} - \frac{\sum_{\mathbf{s}_{N_{j\setminus i}}} s_\mu \, e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}}{\sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}} \times \frac{\sum_{\mathbf{s}_{N_{j\setminus i}}} s_\nu \, e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}}{\sum_{\mathbf{s}_{N_{j\setminus i}}} e^{-\beta H_{N_{j\setminus i}}(\mathbf{s}_{N_{j\setminus i}})}}, \tag{B.29}$$

which we note is temperature dependent. Using the shorthand from Eq. (B.20), D can also be written in the simpler form

$$D_{j\to i,\nu\to\mu} = \langle s_\mu s_\nu \rangle_{0,N_{j\setminus i}} - \langle s_\mu \rangle_{0,N_{j\setminus i}} \langle s_\nu \rangle_{0,N_{j\setminus i}}. \tag{B.30}$$

At the temperature where the magnitude of the leading eigenvalue $\lambda_{\max}$ of J is 1 at the trivial fixed point, the fixed point transitions from being stable to unstable, which corresponds to the phase transition as described in the main text. Thus we can locate the phase transition by evaluating the matrices $\tilde{B}$ and D numerically and using them to compute $|\lambda_{\max}|$. Note that the expectations in Eq. (B.30) do not depend on the values of the messages, so we do not need to perform message passing to calculate them—evaluating the Jacobian and locating the phase transition requires us only to perform the sums over neighborhoods or approximate them using local Monte Carlo.

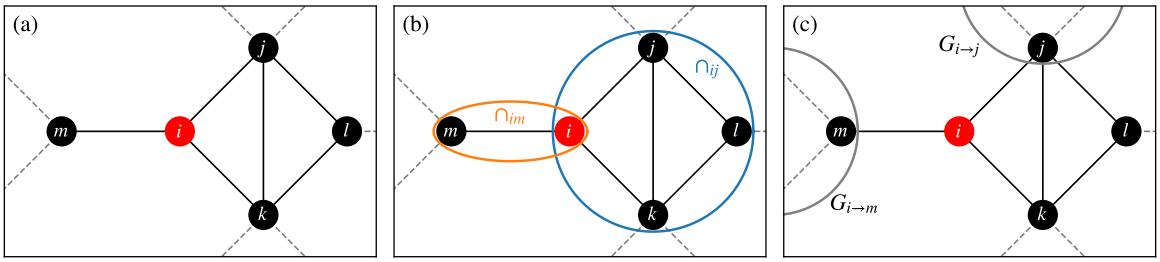

Fig. B.1. Neighborhoods and various related quantities for a node i in an example network. In this example we assume that $r = 2$ is sufficient to capture all primitive cycles and thus that calculations at $r = 2$ are exact. (a) The neighborhood $N_i = N_i^{(2)}$ contains the edges and nodes shown in solid black. (b) At node i there are two distinct intersections, $\cap_{im} = N_i \cap N_m$ and $\cap_{ij} = N_i \cap N_j$. Note that the intersections for all pairs of nodes in $\cap_{ij}$ are identical. For instance in this example we have $\cap_{ij} = \cap_{ik} = \cap_{il} = \cap_{jk} = \cap_{jl} = \cap_{lk}$. (c) The subgraph $G_{i\to j}$ is the connected component to which j belongs after all edges in N_i are removed, and similarly for $G_{i\to m}$.

B.4. Proof of neighborhood-level factorization

In the calculation of the partition function and entropy in Section 4.2.4 of the main text we make use of the factorized form

$$P(\mathbf{s}) = \frac{\prod_{i\in G} P(\mathbf{s}_{N_i})}{\prod_{((i,j))\in G} P(\mathbf{s}_{\cap_{ij}})^{2/|\cap_{ij}|}}\,, \tag{B.31}$$

where $\cap_{ij} = N_i \cap N_j$ and $((i,j))$ are pairs of nodes that are contained in each other's neighborhood, i.e., nodes i and j such that $i \in N_j$ and $j \in N_i$. This form is derived as follows.

Consider Fig. B.1, which illustrates the definition of the sets of nodes we use and their intersections. As shown in panel (b) of the figure, many of the sets are equivalent to one another. Specifically, for any pair $k, l \in \cap_{ij}$ we have $\cap_{kl} = \cap_{ij}$. This allows us

to write

$$P(\mathbf{s}_{\cap_{ij}}) = \left[\prod_{(k,l)\in\cap_{ij}} P(\mathbf{s}_{\cap_{kl}}) \right]^{1/\binom{|\cap_{ij}|}{2}} = \prod_{(k,l)\in\cap_{ij}} P(\mathbf{s}_{\cap_{kl}})^{1/\binom{|\cap_{kl}|}{2}}, \tag{B.32}$$

where the product is over all $\binom{|\cap_{ij}|}{2}$ pairs $\{k,l\} \in \cap_{ij}$. A proof of Eq. (B.31) can then be achieved by induction. Assume that the formula is correct for all networks with fewer than n nodes and no primitive cycles longer than $r+2$. If G is a network with n nodes and no primitive cycles longer than $r+2$ then

$$\begin{aligned} P(\mathbf{s}) &= P(\mathbf{s}_{N_i}) \prod_{j\in N_i} P(\mathbf{s}_{N_j}|\mathbf{s}_{N_i}) P(\mathbf{s}_{G_{i\to j}}|\mathbf{s}_{N_j}) \\ &= P(\mathbf{s}_{N_i}) \prod_{j\in N_i} \frac{P(\mathbf{s}_{N_j})}{P(\mathbf{s}_{\cap_{ij}})} P(\mathbf{s}_{G_{i\to j}}|\mathbf{s}_{N_j\setminus N_i}), \end{aligned} \tag{B.33}$$

where $G_{i\to j}$ denotes the connected subgraph to which j belongs after all edges in N_i have been removed (see Fig. B.1). Since by definition the $G_{i\to j}$ have fewer than n nodes and no primitive cycles longer than $r+2$, Eq. (B.31) is by hypothesis true for these subgraphs, and using (B.32) we have

$$\begin{aligned} P(\mathbf{s}) &= P(\mathbf{s}_{N_i}) \prod_{j\in N_i} \frac{1}{\prod_{(k,l)\in\cap_{ij}} P(\mathbf{s}_{\cap_{kl}})^{1/\binom{|\cap_{kl}|}{2}}} \frac{\prod_{k\in G_{i\to j}} P(\mathbf{s}_{N_k})}{\prod_{(k,l)\in G_{i\to j}} P(\mathbf{s}_{\cap_{kl}})^{2/|\cap_{kl}|}} \\ &= \frac{\prod_{i\in G} P(\mathbf{s}_{N_i})}{\prod_{(i,j)\in G} P(\mathbf{s}_{\cap_{ij}})^{2/|\cap_{ij}|}}. \end{aligned} \tag{B.34}$$

The base case is a graph with a single node, for which (B.31) is trivially true, and hence by induction (B.31) is true for all networks that have no primitive cycles longer than $r+2$.

For the purposes of the calculation presented in Section 4.2.4, Eq. (B.31) can be further simplified by noting that

$$\begin{aligned} P(\mathbf{s}_{N_i}) &= P(s_i) \prod_{j \in N_i} P(\mathbf{s}_{\cap_{ij}} | s_i)^{\frac{1}{|\cap_{ij}|-1}} \\ &= P(s_i) \prod_{j \in N_i} \left[\frac{P(\mathbf{s}_{\cap_{ij}})}{P(s_i)} \right]^{\frac{1}{|\cap_{ij}|-1}}. \end{aligned} \tag{B.35}$$

Substituting this result into (B.31) then yields

$$P(\mathbf{s}) = \prod_{((i,j)) \in G} P(\mathbf{s}_{\cap_{ij}})^{1/\binom{|\cap_{ij}|}{2}} \prod_{(i,j) \in G} P(s_i, s_j)^{W_{ij}} \prod_{i \in G} P(s_i)^{C_i}, \tag{B.36}$$

where

$$W_{ij} = 1 - \sum_{((l,m)) \in G} \frac{1}{\binom{|\cap_{lm}|}{2}} \mathbf{1}_{\{(i,j) \in \cap_{lm}\}} \tag{B.37}$$

and

$$C_i = 1 - \sum_{j \in N_i} \frac{1}{|\cap_{ij}| - 1} - \sum_{j \in N_i^{(0)}} W_{ij}. \tag{B.38}$$

The one- and two-spin marginals $P(s_i)$ and $P(s_i, s_j)$ can be calculated using the message passing methods described in the text, while the intersection marginal $P(\mathbf{s}_{\cap_{ij}})$ is given by

$$P(\mathbf{s}_{\cap_{ij}}) = \frac{1}{Z_{\cap_{ij}}} e^{-\beta H(\mathbf{s}_{\cap_{ij}})} q_{i \leftarrow j}(s_j) \prod_{k \in \cap_{ij} \setminus j} q_{j \leftarrow k}(s_k), \tag{B.39}$$

where $H(\mathbf{s}_{\cap_{ij}})$ denotes the terms of the full Hamiltonian that fall in $\cap_{ij}$ and $Z_{\cap_{ij}}$ is the corresponding normalizing constant.

Equation B.36 is exact when the network contains no primitive cycles longer than $r+2$, in which case $W_{ij} = 0$. When there are longer primitive cycles (and hence Eq. (B.31)

is not exact), the terms $P(s_i, s_j)^{W_{ij}}$ ensure that each edge gets weighted correctly in the factorization.

Appendix C.

Supplementary Material for Chapter 5

In this appendix we provide detailed derivations for multiple results in Chapter 5, as well as a table demonstrating the effect of sample size on the number of modes and plots of the identified modes for additional real example networks.

C.1. Derivation of the description length

The description length is equal to the amount of information needed to transmit the complete set of sampled partitions. We break up the transmission procedure into four separate steps:

1. We transmit S vectors $\mathbf{a}^{(p)}$, one for each $p = 1 \ldots S$. If partition p has n_p non-empty communities, then there are $\binom{n-1}{n_p-1}$ ways to choose the values in the vector $\mathbf{a}^{(p)}$ and hence $\binom{n-1}{n_p-1}$ possible messages that may need to be transmitted to the receiver to communicate $\mathbf{a}^{(p)}$. In binary, our encoding thus requires $\log \binom{n-1}{n_p-1}$ bits, where log denotes the logarithm base 2. (Strictly the number of bits is equal to the smallest integer that is greater than or equal to this num-

ber, but the difference is negligible for large n.) The information required for transmitting all count vectors $\mathbf{a}^{(p)}$ is then

$$L_1 = \sum_{p=1}^{S} \log \binom{n-1}{n_p - 1}. \tag{C.1}$$

This quantity does not depend on the choice of modes or cluster assignments, so we can ignore it when we optimize the total description length of our encoding. It is conceptually important, however, that the $\mathbf{a}^{(p)}$ are transmitted first, as they are needed for constructing efficient encodings for other quantities.

2. Next we transmit the full set of group labels $\hat{\mathbf{g}}^{(k)}$ for each of the mode partitions, exploiting the fact that we now know the label count vector $\mathbf{a}^{(m_k)}$ for each mode. The number of possible sets of group labels consistent with this vector is given by $n!/\prod_{r=1}^{n_{m_k}} a_r^{(m_k)}!$ and hence the number of bits required to transmit a particular set of modes is

$$L_2 = \sum_{k=1}^{K} \log \left(\frac{n!}{\prod_{r=1}^{n_{m_k}} a_r^{(m_k)}!} \right). \tag{C.2}$$

3. For each partition p, we transmit the partition number m_k of the mode to which it belongs. This effectively specifies the clusters themselves. This can be done efficiently by first transmitting the size $c_k = |C_k|$ of each of the K clusters. There are $\binom{S-1}{K-1}$ possible choices such that $\sum_{k=1}^{K} c_k = S$, so it takes $\log \binom{S-1}{K-1}$ bits to transmit any one choice. Then, given the c_k there are $S!/\prod_{k=1}^{K} c_k!$ possible ways to assign the partitions to the clusters, so the total number of bits required to

transmit the cluster labels for all partitions is

$$L_3 = \log \binom{S-1}{K-1} + \log \left(\frac{S!}{\prod_{k=1}^{K} c_k!} \right). \tag{C.3}$$

4. Finally, we transmit the groups labels $\mathbf{g}^{(p)}$ for each individual partition other than the modes, making use of the fact that the modes have already been transmitted. We do this in two steps:

   a) We first transmit the contingency table $\mathbf{t}^{m_k p}$. Since the receiver knows $\mathbf{a}^{(m_k)}$ and $\mathbf{a}^{(p)}$, they also know the row and column sums of $\mathbf{t}^{m_k p}$ because

   $$\sum_r t_{rs}^{m_k p} = a_s^{(p)} \tag{C.4}$$

   and

   $$\sum_s t_{rs}^{m_k p} = a_r^{(m_k)}. \tag{C.5}$$

   If there are $\Omega(m_k, p)$ possible contingency tables with these row and column sums, then it takes $\log \Omega(m_k, p)$ bits to transmit the contingency table $\mathbf{t}^{m_k p}$. Closed-form expressions for $\Omega(m_k, p)$ exist for smaller tables. For larger ones there are good approximations, as described in Ref. [265].

   b) Given the contingency table, the number of partitions consistent with the table is $\prod_{r=1}^{n_{m_k}} [a_r^{(m_k)}! / \prod_{s=1}^{n_p} t_{rs}^{m_k p}!]$ and the number of bits needed to transmit one partition is the log of this number.

The total number of bits required for transmitting the non-mode partitions is

thus

$$L_4 = \sum_{k=1}^{K} \sum_{\substack{p \in C_k \\ p \neq m_k}} \left[\log \prod_{r=1}^{n_{m_k}} \frac{a_r^{(m_k)}!}{\prod_{s=1}^{n_p} t_{rs}^{m_k p}!} + \log \Omega(m_k, p) \right]. \tag{C.6}$$

In practice, the exclusion of the term $p = m_k$ from the sums makes little difference and can be neglected without significantly changing the results, so we will henceforth include this term for notational convenience.

Combining everything, the total description length for the model is

$$L_{\text{total}} = L_1 + L_2 + L_3 + L_4. \tag{C.7}$$

For our purposes it is convenient to normalize this as description length per sample, which gives

$$\mathcal{L}_{\text{total}} = \frac{1}{S}(L_1 + L_2 + L_3 + L_4). \tag{C.8}$$

We can convert this quantity to more familiar language by using Stirling's approximation, whose leading terms for base-2 logarithms can be written in the form

$$\log x! \simeq x \log x - \frac{x}{\ln 2}. \tag{C.9}$$

Dropping the term L_1 from Eq. C.8 as discussed previously, we then have

$$\begin{aligned} \mathcal{L}_{\text{total}} \simeq \frac{n}{S} \sum_{k=1}^{K} H(\hat{\mathbf{g}}^{(k)}) + H(\mathbf{c}) + \frac{n}{S} \sum_{k=1}^{K} \sum_{p \in C_k} H_{\text{mod}}(\mathbf{g}^{(p)} | \hat{\mathbf{g}}^{(k)}) \\ + \frac{S-1}{S} \log(S-1) - \frac{S-K}{S} \log(S-K) - \frac{K-1}{S} \log(K-1). \end{aligned} \tag{C.10}$$

Assuming $S \gg K$ (but not assuming, crucially, that K remains constant as $S \to \infty$), we can drop the last three terms in Eq. C.10, giving the form:

$$\mathcal{L}_{\text{total}} \simeq \frac{n}{S} \sum_{k=1}^{K} H(\hat{\mathbf{g}}^{(k)}) + H(\mathbf{c}) + \frac{n}{S} \sum_{k=1}^{K} \sum_{p \in C_k} H_{\text{mod}}(\mathbf{g}^{(p)} | \hat{\mathbf{g}}^{(k)}), \tag{C.11}$$

up to an additive constant.

C.2. Number of clusters

Here we demonstrate that the optimal value of K in the penalized description length is asymptotically constant as the number of samples S grows. For the purposes of our argument we assume that all partitions p have the same number of groups n_p, that the number of nodes n is fixed and $n \gg n_p$, and that the cluster sizes c_k are approximately equal. We do not neglect the last three terms in Eq. C.10 as we did previously, for a more careful treatment.

In terms of S, K, n, and n_p, the leading order scaling of each of the terms in Eq. C.10, along with the linear penalty term $+\lambda K$, is

$$\begin{aligned} \mathcal{L}(S, K) \sim \frac{Kn}{S} \log n_p + \log K + \frac{n(S-K)}{S} \tilde{H}_{\text{mod}}(K) \\ + \frac{S-1}{S} \log(S-1) - \frac{S-K}{S} \log(S-K) - \frac{K-1}{S} \log(K-1) + \lambda K, \end{aligned} \tag{C.12}$$

where $\widetilde{H}_{\text{mod}}(K)$ is a typical scale for $H_{\text{mod}}(\mathbf{g}^{(p)} | \hat{\mathbf{g}}^{(k)})$. In general $\widetilde{H}_{\text{mod}}(K)$ is a decreasing function of K, since a larger number of clusters allows partitions to be assigned to closer modes. We ignore the $\log \Omega/n$ contribution to H_{mod}, as it scales like $n_p^2 \log n/n$ [265] and can be neglected by comparison with the $\mathrm{O}(\log n_p)$ contribution

from the standard conditional entropy when $n \gg n_p$.

For fixed S, a local minimum of Eq. C.12 with respect to K occurs at the first value of K for which

$$\mathcal{L}(S, K+1) - \mathcal{L}(S, K) > 0. \tag{C.13}$$

To demonstrate that the optimal value of K remains constant as S increases, we let $S \to \infty$ in Eq. C.12 and show that we can always satisfy Eq. C.13 with a finite value of K that is independent of S. Letting $S \to \infty$ in Eq. C.12 with K constant and substituting into Eq. C.13 gives

$$\log(1 + 1/K) + \lambda + n[\widetilde{H}_{\text{mod}}(K+1) - \widetilde{H}_{\text{mod}}(K)] > 0, \tag{C.14}$$

where we have discarded terms of order $\log S/S$ and smaller. Rearranging gives

$$\widetilde{H}_{\text{mod}}(K) - \widetilde{H}_{\text{mod}}(K+1) < \frac{\lambda}{n} + \frac{1}{n}\log(1 + 1/K). \tag{C.15}$$

Because $H_{\text{mod}}(K)$ is a decreasing function of K, this inequality will always be satisfied for some constant K, since $H_{\text{mod}}(K) - H_{\text{mod}}(K+1)$ approaches 0 from above and the right-hand side is bounded below by the strictly positive constant λ/n. Thus the optimal value of K in Eq. C.12 is asymptotically constant as S grows.

Note that we cannot make the same argument for the unpenalized description length of Eq. C.7. In that case the inequality analogous to Eq. C.15 is

$$\widetilde{H}_{\text{mod}}(K) - \widetilde{H}_{\text{mod}}(K+1) < \frac{1}{n}\log(1 + 1/K), \tag{C.16}$$

but the right-hand side of this expression goes to zero as K becomes large, so we cannot guarantee there is a finite value of K that satisfies the inequality. In practice, we find that this inequality is not satisfied in many test networks, the optimal K growing monotonically with S.

In Table C.1, we display the optimal number of clusters K for various sample sizes S and $\lambda = 0, 1$, for the networks shown in Chapter 5 and this appendix. We can see that for $\lambda = 0$ the number of clusters grows substantially with the sample size S, whereas with $\lambda = 1$ it remains nearly constant for most of the examples. The biggest exception is the network science collaboration network, which does differ by a few clusters as we increase S but not by many. This illustrates that, despite the scaling in Eq. C.15 being only approximate for $S \to \infty$, the constraint λK is effective in practical applications for reducing the effect of the sample size on the number of clusters.

C.3. Additional example applications

In Fig. C.1 we show two additional example applications of our method. Figure C.1A shows a network of collaborations among researchers in the field of network science [289], which exhibits highly multimodal community structure. In a manner reminiscent of the artificial network of cliques in Fig. 2C, this network consists of many small, tightly connected groups of nodes, which can be arranged in various ways to form plausible community divisions. As we might expect, the modes identified for this network appear to be comprised of a few of these possible arrangements.

In Fig. C.1B we show the modes of a network of associations among terrorists involved in the 2004 Madrid train bombing [290]. In this case, we see that the community structure in the upper region of the network is uncertain, resulting in two

Table C.1: Number of clusters K for various sample sizes S, and $\lambda = 0, 1$, for example networks.

Network	Number of samples S	Optimal K, $\lambda = 0$	Optimal K, $\lambda = 1$
Planted partition	100	1	1
	1000	1	1
	10000	3	1
Nested SBM	100	2	2
	1000	2	2
	10000	8	2
Cliques	100	2	2
	1000	10	2
	10000	29	2
Political books	100	2	2
	1000	8	2
	10000	25	2
AddHealth	100	2	2
	1000	8	3
	10000	19	3
Chicago	100	1	1
	1000	3	3
	10000	14	3
Collaborations	100	2	2
	1000	8	4
	10000	26	6
Terrorists	100	3	2
	1000	6	2
	10000	17	2

substantially distinct community divisions appearing as modes.

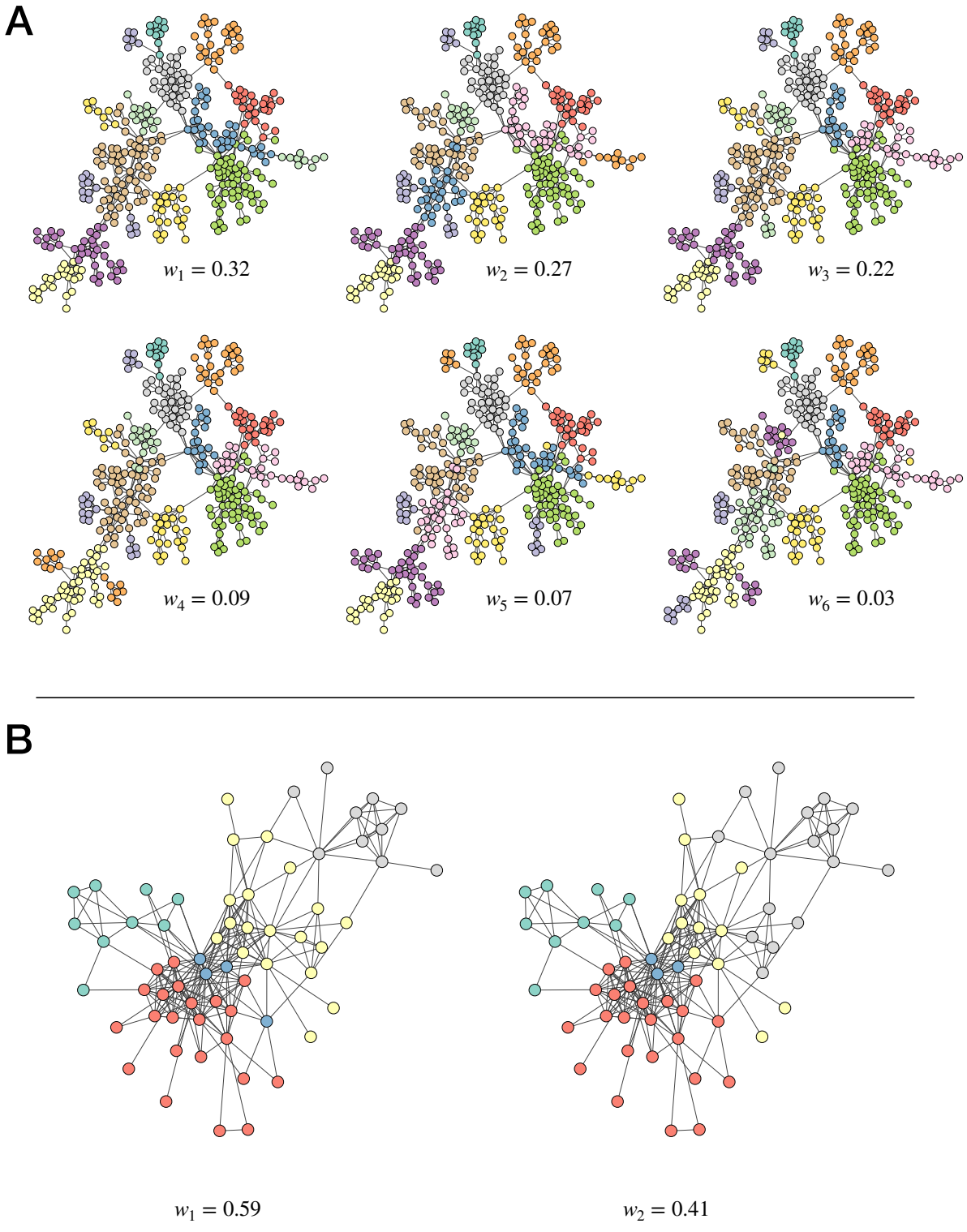

Fig. C.1. Representative modes and their corresponding weights for two additional real-world example networks, identified by minimizing the penalized description length with $\lambda = 1$ for $10,000$ community partition samples. (A) Collaboration network among network scientists [289]. (B) Network of terrorist associations [290].

Bibliography

1. Newman, M. *Networks* 2nd ed. (Oxford University Press, Oxford, 2018).
2. Barabási, A.-L. *Network Science* (Cambridge University Press, Cambridge, 2016).
3. Barrat, A., Barthélemy, M. & Vespignani, A. *Dynamical Processes on Complex Networks* (Cambridge University Press, Cambridge, 2008).
4. Boccaletti, S., Latora, V., Moreno, Y., Chavez, M. & Hwang, D.-U. Complex networks: Structure and dynamics. *Physics Reports* **424,** 175–308 (2006).
5. Dorogovtsev, S. *Lectures on Complex Networks* (Oxford University Press, Oxford, 2010).
6. Newman, M. E. J. The structure and function of complex networks. *SIAM Review* **45,** 167–256 (2003).
7. Newman, M. E. J., Barabási, A.-L. & Watts, D. J. *The Structure and Dynamics of Networks* (Princeton University Press, Princeton, 2006).
8. Bollobás, B. *Random Graphs* (Academic Press, New York, 1985).
9. Caldarelli, G. *Scale-Free Networks* (Oxford University Press, Oxford, 2007).
10. Cohen, R. & Havlin, S. *Complex Networks: Structure, Stability and Function* (Cambridge University Press, Cambridge, 2010).
11. Easley, D. & Kleinberg, J. *Networks, Crowds, and Markets* (Cambridge University Press, Cambridge, 2010).
12. Albert, R. & Barabási, A.-L. Statistical mechanics of complex networks. *Reviews of Modern Physics* **74,** 47–97 (2002).
13. Jackson, M. O. *Social and Economic Networks* (Princeton University Press, Princeton, 2008).
14. Crane, H. *Probabilistic Foundations of Statistical Network Analysis* (CRC Press, Boca Raton, 2018).
15. Bassett, D. S. & Sporns, O. Network neuroscience. *Nature Neuroscience* **20,** 353–364 (2017).
16. Sporns, O. Contributions and challenges for network models in cognitive neuroscience. *Nature Neuroscience* **17,** 652–660 (2014).

17. Gale, D. M. & Kariv, S. Financial networks. *American Economic Review* **97,** 99–103 (2007).

18. Boginski, V., Butenko, S. & Pardalos, P. M. Statistical analysis of financial networks. *Computational Statistics & Data Analysis* **48,** 431–443 (2005).

19. Schweitzer, F. *et al.* Economic networks: The new challenges. *Science* **325,** 422–425 (2009).

20. Collar, A., Coward, F., Brughmans, T. & Mills, B. J. Networks in archaeology: Phenomena, abstraction, representation. *Journal of Archaeological Method and Theory* **22,** 1–32 (2015).

21. Knappett, C. *Network Analysis in Archaeology: New Approaches to Regional Interaction* (Oxford University Press, Oxford, 2013).

22. Feng, S. & Law, N. Mapping artificial intelligence in education research: A network-based keyword analysis. *International Journal of Artificial Intelligence in Education* **31,** 277–303 (2021).

23. Calvó-Armengol, A., Patacchini, E. & Zenou, Y. Peer effects and social networks in education. *The Review of Economic Studies* **76,** 1239–1267 (2009).

24. Ward, M. D., Stovel, K. & Sacks, A. Network analysis and political science. *Annual Review of Political Science* **14,** 245–264 (2011).

25. Knoke, D. *Political Networks: The Structural Perspective* (Cambridge University Press, Cambridge, 1994).

26. Barabási, A.-L., Gulbahce, N. & Loscalzo, J. Network medicine: A network-based approach to human disease. *Nature Reviews Genetics* **12,** 56–68 (2011).

27. Pawson, T. & Linding, R. Network medicine. *FEBS Letters* **582,** 1266–1270 (2008).

28. Moreno, J. L. *Who Shall Survive?* (Beacon House, Beacon, NY, 1934).

29. Travers, J. & Milgram, S. An experimental study of the small world problem. *Sociometry* **32,** 425–443 (1969).

30. Granovetter, M. The strength of weak ties. *American Journal of Sociology* **78,** 1360–1380 (1973).

31. Lazer, D. *et al.* Life in the network: the coming age of computational social science. *Science* **323,** 721 (2009).

32. Huberman, B. A. & Adamic, L. A. Growth dynamics of the World-Wide Web. *Nature* **401,** 131 (1999).

33. Ugander, J., Karrer, B., Backstrom, L. & Marlow, C. *The anatomy of the Facebook social graph* Preprint arxiv:1111.4503 (2011).

34. Kivelä, M. *et al.* Multilayer networks. *Journal of Complex Networks* **2,** 203–271 (2014).

35. Boccaletti, S. *et al.* The structure and dynamics of multilayer networks. *Physics Reports* **544,** 1–122 (2014).

36. De Domenico, M., Granell, C., Porter, M. A. & Arenas, A. The physics of multilayer networks. *Nature Physics* **12,** 901–906 (2016).

37. Ghoshal, G., Zlatic, V., Caldarelli, G. & Newman, M. E. J. Random hypergraphs and their applications. *Physical Review E* **79,** 066118 (2009).

38. Horak, D., Maletić, S. & Rajković, M. Persistent homology of complex networks. *Journal of Statistical Mechanics: Theory and Experiment* **2009,** p03034 (2009).

39. Iñiguez, G., Battiston, F. & Karsai, M. Bridging the gap between graphs and networks. *Communications Physics* **3,** 1–5 (2020).

40. *The Art of Computer Programming* (ed Knuth, D.) (Pearson Education, Boston, 1968).

41. Milgram, S. The small world problem. *Psychology Today* **2,** 60–67 (1967).

42. Watts, D. J. & Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature* **393,** 440–442 (1998).

43. Cohen, R., Erez, K., ben-Avraham, D. & Havlin, S. Resilience of the Internet to random breakdowns. *Physical Review Letters* **85,** 4626–4628 (2000).

44. Cohen, R. & Havlin, S. Scale-free networks are ultrasmall. *Physical Review Letters* **90,** 058701 (2003).

45. Pastor-Satorras, R. & Vespignani, A. in *Handbook of Graphs and Networks* (eds Bornholdt, S. & Schuster, H. G.) (Wiley-VCH, Berlin, 2003).

46. Cohen, R., Havlin, S. & ben-Avraham, D. Efficient immunization strategies for computer networks and populations. *Physical Review Letters* **91,** 247901 (2003).

47. Barabási, A.-L. & Albert, R. Emergence of scaling in random networks. *Science* **286,** 509–512 (1999).

48. Barabási, A.-L., Albert, R. & Jeong, H. Scale-free characteristics of random networks: The topology of the World Wide Web. *Physica A* **281,** 69–77 (2000).

49. Price, D. J. d. Networks of scientific papers. *Science* **149,** 510–515 (1965).

50. Price, D. J. d. A general theory of bibliometric and other cumulative advantage processes. *Journal of the Association for Information Science and Technology* **27,** 292–306 (1976).

51. Broido, A. D. & Clauset, A. Scale-free networks are rare. *Nature Communications* **10,** 1–10 (2019).

52. Voitalov, I., van der Hoorn, P., van der Hofstad, R. & Krioukov, D. Scale-free networks well done. *Physical Review Research* **1,** 033034 (2019).

53. Holme, P. Rare and everywhere: Perspectives on scale-free networks. *Nature Communications* **10,** 1–3 (2019).

54. Chung, F. R. K. *Spectral Graph Theory CBMS Regional Conference Series in Mathematics* **92** (American Mathematical Society, Providence, RI, 1997).

55. Newman, M. E. J. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences* **98,** 404–409 (2001).

56. Newman, M. E. J. & Park, J. Why social networks are different from other types of networks. *Physical Review E* **68,** 036122 (2003).

57. Newman, M. E. J. Properties of highly clustered networks. *Physical Review E* **68,** 026121 (2003).

58. Miller, J. C. Percolation and epidemics in random clustered networks. *Physical Review E* **80,** 020901 (2009).

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

60. Fosdick, B. K., Larremore, D. B., Nishimura, J. & Ugander, J. Configuring random graph models with fixed degree sequences. *Siam Review* **60,** 315–355 (2018).

61. Chen, M., Kuzmin, K. & Szymanski, B. K. Community detection via maximization of modularity and its variants. *IEEE Transactions on Computational Social Systems* **1,** 46–65 (2014).

62. Stegehuis, C., Hofstad, R. V. D. & Leeuwaarden, J. S. V. Epidemic spreading on complex networks with community structures. *Scientific Reports* **6,** 1–7 (2016).

63. Peixoto, T. P. Reconstructing networks with unknown and heterogeneous errors. *Physical Review X* **8,** 041011 (2018).

64. Gerlach, M., Peixoto, T. P. & Altmann, E. G. A network approach to topic models. *Science Advances* **4,** eaaq1360 (2018).

65. Chung, F. & Lu, L. Connected components in random graphs with given degree sequences. *Annals of Combinatorics* **6,** 125–145 (2002).

66. Holme, P. & Kim, B. J. Growing scale-free networks with tunable clustering. *Physical Review E* **65,** 026107 (2002).

67. Holland, P. W., Laskey, K. B. & Leinhardt, S. Stochastic blockmodels: First steps. *Social Networks* **5,** 109–137 (1983).

68. Harris, J. K. *An Introduction to Exponential Random Graph Modeling* (Sage Publications, Thousand Oaks, CA, 2013).

69. Morel-Balbi, S. & Peixoto, T. P. Null models for multioptimized large-scale network structures. *Physical Review E* **102,** 032306 (2020).

70. Koevering, K. V., Benson, A. & Kleinberg, J. *Random Graphs with Prescribed K-Core Sequences: A New Null Model for Network Analysis* in *Proceedings of the Web Conference 2021* (Association for Computing Machinery, New York, 2021), 367–378.

71. Erdős, P. & Rényi, A. On random graphs. *Publicationes Mathematicae* **6,** 290–297 (1959).

72. Gilbert, E. N. Random graphs. *Annals of Mathematical Statistics* **30,** 1191–1141 (1959).

73. Merton, R. K. The Matthew effect in science. *Science* **159,** 56–63 (1968).

74. Airoldi, E. M., Blei, D. M., Fienberg, S. E. & Xing, E. P. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research* **9,** 1981–2014 (2008).

75. Peixoto, T. P. Hierarchical block structures and high-resolution model selection in large networks. *Physical Review X* **4,** 011047 (2014).

76. Aicher, C., Jacobs, A. Z. & Clauset, A. Learning latent block structure in weighted networks. *Journal of Complex Networks* **3,** 221–248 (2025).

77. Karrer, B. & Newman, M. E. J. Stochastic blockmodels and community structure in networks. *Physical Review E* **83,** 016107 (2011).

78. Gelman, A. *et al. Bayesian Methods for Data Analysis* 3rd ed. (CRC Press, New York, 2014).

79. Newman, M. E. J. Network structure from rich but noisy data. *Nature Physics* **14,** 542–545 (2018).

80. Young, J.-G., Petri, G. & Peixoto, T. P. Hypergraph reconstruction from network data. *Communications Physics* **4,** 135 (2021).

81. Peixoto, T. P. & Gauvin, L. Change points, memory and epidemic spreading in temporal networks. *Scientific Reports* **8,** 15511 (2018).

82. Peixoto, T. P. Latent Poisson models for networks with heterogeneous density. *Physical Review E* **102,** 012309 (2020).

83. Young, J.-G., Kirkley, A. & Newman, M. E. J. *Clustering of heterogeneous populations of networks* Preprint arxiv:2107.07489 (2021).

84. Young, J.-G. *et al.* Phase Transition in the Recoverability of Network History. *Physical Review X* **9,** 041056 (2019).

85. Cantwell, G. T., St-Onge, G. & Young, J.-G. Inference, Model Selection, and the Combinatorics of Growing Trees. *Physical Review Letters* **126,** 038301 (2021).

86. Kirkley, A., Cantwell, G. T. & Newman, M. E. J. Balance in signed networks. *Physical Review E* **99,** 012320 (2019).

87. Kirkley, A. Information theoretic network approach to socioeconomic correlations. *Physical Review Research* **2,** 043212 (2020).

88. Kirkley, A., Cantwell, G. & Newman, M. E. J. Belief propagation for networks with loops. *Science Advances* **7,** eabf1211 (2021).

89. Kirkley, A. & Newman, M. E. J. *Representative community divisions of networks* Preprint arXiv:2105.04612 (2021).

90. Harary, F. On the notion of balance of a signed graph. *Michigan Mathematical Journal* **2,** 143–146 (1953).

91. Wasserman, S. & Faust, K. *Social Network Analysis* (Cambridge University Press, Cambridge, 1994).

92. Facchetti, G., Iacono, G. & Altafini, C. Computing global structural balance in large-scale signed social networks. *Proceedings of the National Academy of Sciences* **108,** 20953–20958 (2011).

93. Estrada, E. & Benzi, M. *Are social networks really balanced?* Preprint arxiv:1406.2132 (2014).

94. Estrada, E. & Benzi, M. Walk-based measure of balance in signed networks: Detecting lack of balance in social networks. *Physical Review E* **90,** 042802 (2014).

95. Cartwright, D. & Harary, F. Structural balance: A generalization of Heider's theory. *Psychological Review* **63,** 277 (1956).

96. Davis, J. A. Clustering and structural balance in graphs. *Human Relations* **20,** 181–187 (1967).

97. Aref, S. & Wilson, M. C. *Measuring partial balance in signed networks* Preprint arxiv:1509.04037 (2015).

98. Belaza, A. *et al.* Statistical physics of balance theory. *PLOS One* **12,** e0183696 (2017).

99. Belaza, A. *et al.* Social stability and extended social balance-quantifying the role of inactive links in social networks. *Physica A* **518,** 270–284 (2019).

100. Marvel, S. A., Strogatz, S. H. & Kleinberg, J. M. Energy landscape of social balance. *Physical Review Letters* **103,** 198701 (2009).

101. Altafini, C. Dynamics of opinion forming in structurally balanced social networks. *PLOS One* **7,** e38135 (2012).

102. Zheng, X., Zeng, D. & Wang, F. Y. Social balance in signed networks. *Information Systems Frontiers* **17,** 1077–1095 (2015).

103. Acharya, B. D. Spectral criterion for cycle balance in networks. *Journal of Graph Theory* **4,** 1–11 (1980).

104. Anchuri, P. & Magdon-Ismail, M. *An information-theoretic external cluster-validity measure* in *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)* (Institute of Electrical and Electronics Engineers, New York, 2012), 235–242.

105. Kunegis, J. *et al. Spectral analysis of signed graphs for clustering, prediction and visualization* in *Proceedings of the 2010 SIAM International Conference on Data Mining* (Society for Industrial and Applied Mathematics, Philadelphia, 2010), 559–570.

106. Harary, F. On the measurement of structural balance. *Systems Research and Behavioral Science* **4,** 316–323 (1959).

107. Norman, R. Z. & Roberts, F. S. A derivation of a measure of relative balance for social structures and a characterization of extensive ratio systems. *Journal of Mathematical Psychology* **9,** 66–91 (1972).

108. Chiang, K.-Y., Nataraja, N., Tewari, A. & Dhillon, I. S. *Exploiting longer cycles for link prediction in signed networks* in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management* (Association of Computing Machinery, New York, 2011), 1157–1162.

109. Chiang, K.-Y., Hsieh, C.-J., Natarajan, N., Dhillon, I. S. & Tewari, A. Prediction and clustering in signed networks: A local to global perspective. *Journal of Machine Learning Research* **15,** 1177–1213 (2014).

110. Singh, R. & Adhikari, B. Measuring the balance of signed networks and its application to sign prediction. *Journal of Statistical Mechanics: Theory and Experiment* **2017,** 063302 (2017).

111. Latora, V., Nicosia, V. & Russo, G. *Complex Networks: Principles, Methods and Applications* (Cambridge University Press, Cambridge, 2017).

112. Guha, R., Kumar, R., Raghavan, P. & Tomkins, A. *Propagation of trust and distrust* in *Proceedings of the 13th International Conference on the World Wide Web (WWW 2004)* (Association of Computing Machinery, New York, 2004), 403–412.

113. Mouttapa, M., Valente, T., Gallaher, P., Rohrbach, L. A. & Unger, J. B. Social network predictors of bullying and victimization. *Adolescence* **39,** 315–335 (2004).

114. Xia, L., Yuan, Y. C. & Gay, G. Exploring negative group dynamics: Adversarial network, personality, and performance in project groups. *Management Communication Quarterly* **23,** 32–62 (2009).

115. Feng, D., Altmeyer, R., Stafford, D., Christakis, N. A. & Zhou, H. H. Testing for balance in social networks. *Journal of the American Statistical Association* **0,** 1–19 (2020).

116. Izmirlioglu, A. The Correlates of War dataset. *Journal of World-Historical Information* **4** (2017).

117. De Bunt, G. G. V., Duijn, M. A. V. & Snijders, T. A. Friendship networks through time: An actor-oriented dynamic statistical network model. *Computational & Mathematical Organization Theory* **5,** 167–192 (1999).

118. Lerner, J. Structural balance in signed networks: Separating the probability to interact from the tendency to fight. *Social Networks* **45,** 66–77 (2016).

119. Leskovec, J., Huttenlocher, D. & Kleinberg, J. *Predicting positive and negative links in online social networks* in *Proceedings of the 19th International Conference on World Wide Web* (Association of Computing Machinery, New York, 2010), 641–650.

120. Yang, S.-H., Smola, A. J., Long, B., Zha, H. & Chang, Y. *Friend or frenemy? Predicting signed ties in social networks* in *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Association of Computing Machinery, New York, 2012), 555–564.

121. Squartini, T., Caldarelli, G., Cimini, G., Gabrielli, A. & Garlaschelli, D. Reconstruction methods for networks: The case of economic and financial systems. *Physics Reports* **757,** 1–47 (2018).

122. Hager, W. W. Updating the inverse of a matrix. *SIAM Review* **31,** 221–239 (1989).

123. Strang, G. *Introduction to Linear Algebra* (Wellesley Cambridge Press, Wellesley, MA, 2009).

124. Danon, L., Duch, J., Diaz-Guilera, A. & Arenas, A. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment* **2005,** P09008 (2005).

125. Johnston, R. J. *Political, Electoral, and Spatial Systems: An Essay in Political Geography* (Clarendon Press, Oxford, 1979).

126. P. Elliot, Wakefield, J. C., Best, N. G. & Briggs, D. J. *Spatial Epidemiology: Methods and Applications* (Oxford University Press, Oxford, 2000).

127. Walker, R. E., Keane, C. R. & Burke, J. G. Disparities and access to healthy food in the United States: A review of food deserts literature. *Health & Place* **16,** 876–884 (2010).

128. beckmann1985spatial. *Spatial Economics: Density, Potential, and Flow* (North Holland Publishing, Amsterdam, 1985).

129. Reardon, S. F. & D. OSullivan. Measures of spatial segregation. *Sociological Methodology* **34,** 121–162 (2004).

130. Rey, S. J. Spatial analysis of regional income inequality. *Spatially Integrated Social Science* **1,** 280–299 (2004).

131. Brown, J. H., Mehlman, D. W. & Stevens, G. C. Spatial variation in abundance. *Ecology* **76,** 2028–2043 (1995).

132. Bettencourt, L. M., J. Lobo, D. Helbing, C. Kühnert & West, G. B. Growth, innovation, scaling, and the pace of life in cities. *Proceedings of the National Academy of Sciences* **104,** 7301–7306 (2007).

133. Stouffer, S. A. Intervening opportunities and competing migrants. *Journal of Regional Science* **2,** 1–26 (1960).

134. Gastner, M. T. & Newman, M. E. J. Diffusion-based method for producing density equalizing maps. *Proceedings of the National Academy of Sciences* **101,** 7499–7504 (2004).

135. Holt, J. B., C. Lo & Hodler, T. W. Dasymetric estimation of population density and areal interpolation of census data. *Cartography and Geographic Information Science* **31,** 103–121 (2004).

136. Gehlke, C. E. & K. Biehl. Certain effects of grouping upon the size of the correlation coefficient in census tract material. *Journal of the American Statistical Association* **29,** 169–170 (1934).

137. Turner, M. G., O'Neill, R. V., Gardner, R. H. & Milne, B. T. Effects of changing spatial scale on the analysis of landscape pattern. *Landscape Ecology* **3,** 153–162 (1989).

138. C. Flint & Taylor, P. J. *Political Geography: World-economy, Nation-state, and Locality* (Pearson Education, London, 2007).

139. Duncan, O. D. & B. Duncan. A methodological analysis of segregation indexes. *American Sociological Review* **20,** 210–217 (1955).

140. J. Walsh & OKelly, M. E. An information theoretic approach to measurement of spatial inequality. *Economic and Social Review* **10,** 267–286 (1979).

141. P. S. Chodrow. Structure and information in spatial segregation. *Proceedings of the National Academy of Sciences* **114,** 11591–11596 (2017).

142. J. Lin. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory* **37,** 145–151 (1991).

143. M. Barthélemy. Spatial networks. *Physics Reports* **499,** 1–101 (2011).

144. A. Kirkley, H. Barbosa, M. Barthelemy & G. Ghoshal. From the betweenness centrality in street networks to structural invariants in random planar graphs. *Nature Communications* **9,** 1–12 (2018).

145. Clark, W. A., E. Anderson, J. Östh & B. Malmberg. A multiscalar analysis of neighborhood composition in Los Angeles, 2000–2010: A location-based approach to segregation and diversity. *Annals of the Association of American Geographers* **105,** 1260–1284 (2015).

146. M. Olteanu, J. Randon-Furling & Clark, W. A. Segregation through the multiscalar lens. *Proceedings of the National Academy of Sciences* **116,** 12250–12254 (2019).

147. M. Batty. Spatial entropy. *Geographical Analysis* **6,** 1–31 (1974).

148. Vaz, E. & Bandur, D. *Merging Entropy in Self-Organisation: A Geographical Approach* in *Resilience and Regional Dynamics. Advances in Spatial Science (The Regional Science Series)* (Springer, Cham, 2018), 171–186.

149. Bureau, U. C. *American Factfinder* tech. rep. US Department of Commerce, Economics and Statistics Administration (2004).

150. Bureau, U. C. *American community survey 5-year data (2009-2018)* `https://www.census.gov/data/developers/data-sets/acs-5year.html`.

151. Houstoun Jr, L. O. Neighborhood change and city policy. *Urban Land* **35,** 159–170 (1976).

152. N. Krieger. A century of census tracts: Health & the body politic (1906–2006). *Journal of Urban Health* **83,** 355–361 (2006).

153. McDonald, J. B. & M. Ransom. *The generalized beta distribution as a model for the distribution of income: Estimation of related measures of inequality* in *Modeling Income Distributions and Lorenz Curves* (Springer, New York, 2008), 147–166.

154. Von Hippel, P. T., Hunter, D. J. & M. Drown. Better estimates from binned income data: Interpolated CDFs and mean-matching. *Sociological Science* **4,** 641–655 (2017).

155. Bureau, U. C. *Tiger/line shapefiles* 2019.

156. J. Briët & P. Harremoës. Properties of classical and quantum Jensen-Shannon divergence. *Physical Review A* **79,** 052311 (2009).

157. S. Itzkovitz, E. Hodis & E. Segal. Overlapping codes within protein-coding sequences. *Genome Research* **20,** 1582–1589 (2010).

158. S. Klingenstein, T. Hitchcock & S. DeDeo. The civilizing process in London's Old Bailey. *Proceedings of the National Academy of Sciences* **111,** 9419–9424 (2014).

159. Cover, T. M. & Thomas, J. A. *Elements of Information Theory* (John Wiley & Sons, Hoboken, 2012).

160. Y. Rubner, C. Tomasi & Guibas, L. J. *A metric for distributions with applications to image databases* in *Sixth International Conference on Computer Vision* (Institute of Electrical and Electronics Engineers, Bombay, 1998), 59–66.

161. M. Davis & P. Peebles. A survey of galaxy redshifts. V. The two-point position and velocity correlations. *The Astrophysical Journal* **267,** 465–482 (1983).

162. B. Ganapathisubramani, N. Hutchins, W. Hambleton, Longmire, E. K. & I. Marusic. Investigation of large-scale coherence in a turbulent boundary layer using two-point correlations. *Journal of Fluid Mechanics* **524,** 57–80 (2005).

163. Y. Kagan & L. Knopoff. Spatial distribution of earthquakes: the two-point correlation function. *Geophysical Journal International* **62,** 303–320 (1980).

164. D. Rybski, Rozenfeld, H. D. & Kropp, J. P. Quantifying long-range correlations in complex networks beyond nearest neighbors. *Europhysics Letters* **90,** 28002 (2010).

165. M. Mayo, A. Abdelzaher & P. Ghosh. Long-range degree correlations in complex networks. *Computational Social Networks* **2,** 4 (2015).

166. Y. Fujiki, T. Takaguchi & K. Yakubo. General formulation of long-range degree correlations in complex networks. *Physical Review E* **97,** 062308 (2018).

167. Gautheir, T. D. Detecting trends using Spearman's rank correlation coefficient. *Environmental Forensics* **2,** 359–362 (2001).

168. Kahl, J. A. & Davis, J. A. A comparison of indexes of socioeconomic status. *American Sociological Review* **20,** 317–325 (1955).

169. Lawson, E. D. & Boek, W. E. Correlations of indexes of families' socioeconomic status. *Social Forces* **39,** 149–152 (1960).

170. J. Ludwig, Ladd, H. F., Duncan, G. J., J. Kling & OŔegan, K. M. *Urban poverty and educational outcomes* in *Brookings-Wharton Papers on Urban Affairs* (Brookings Institution Press, Washington D.C., 2001), 147–201.

171. S. Moller, Alderson, A. S. & F. Nielsen. Changing patterns of income inequality in US counties, 1970–2000. *American Journal of Sociology* **114,** 1037–1101 (2009).

172. L. Bettencourt & G. West. A unified theory of urban living. *Nature* **467,** 912–913 (2010).

173. Redding, S. J. & E. Rossi-Hansberg. Quantitative spatial economics. *Annual Review of Economics* **9,** 21–58 (2017).

174. Y. Zenou & N. Boccard. Racial discrimination and redlining in cities. *Journal of Urban Economics* **48,** 260–285 (2000).

175. S. Van Nieuwerburgh & Weill, P.-O. Why has house price dispersion gone up? *The Review of Economic Studies* **77,** 1567–1606 (2010).

176. Dwyer, R. E. Expanding homes and increasing inequalities: US housing development and the residential segregation of the affluent. *Social Problems* **54,** 23–46 (2007).

177. Davis, M. A. & Palumbo, M. G. The price of residential land in large US cities. *Journal of Urban Economics* **63,** 352–384 (2008).

178. F. Wang, M. Wen & Y. Xu. Population-adjusted street connectivity, urbanicity and risk of obesity in the US. *Applied Geography* **41,** 1–14 (2013).

179. M. Doussard, J. Peck & N. Theodore. After deindustrialization: Uneven growth and economic inequality in "postindustrial" Chicago. *Economic Geography* **85,** 183–207 (2009).

180. M. Kassen. A promising phenomenon of open data: A case study of the Chicago open data project. *Government Information Quarterly* **30,** 508–513 (2013).

181. S. Sobolevsky, R. Campari, A. Belyi & C. Ratti. General optimization technique for high-quality community detection in complex networks. *Physical Review E* **90,** 012811 (2014).

182. Nelson, G. D. & A. Rae. An economic geography of the United States: From commutes to megaregions. *PLOS One* **11,** e0166083 (2016).

183. Y. Liu *et al.* Social sensing: A new approach to understanding our socioeconomic environments. *Annals of the Association of American Geographers* **105,** 512–530 (2015).

184. Assunção, R. M., Neves, M. C., G. Câmara & C. da Costa Freitas. Efficient regionalization techniques for socioeconomic geographical units using minimum spanning trees. *International Journal of Geographical Information Science* **20,** 797–811 (2006).

185. Kondor, R. I. & J. Lafferty. *Diffusion kernels on graphs and other discrete structures* in *Proceedings of the 19th International Conference on Machine Learning* (Morgan Kaufmann, San Francisco, 2002), 315–322.

186. J. Reichardt & S. Bornholdt. Statistical mechanics of community detection. *Physical Review E* **74,** 016110 (2006).

187. Blondel, V. D., Guillaume, J.-L., R. Lambiotte & E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* **2008,** P10008 (2008).

188. Jain, A. K., Murty, M. N. & Flynn, P. J. Data clustering: a review. *ACM Computing Surveys (CSUR)* **31,** 264–323 (1999).

189. Vinh, N. X., J. Epps & J. Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *The Journal of Machine Learning Research* **11,** 2837–2854 (2010).

190. Kiss, I. Z., Miller, J. C. & Simon, P. L. *Mathematics of Epidemics on Networks* (Springer International Publishing, Berlin, 2017).

191. Carreras, B. A., Lynch, V. E., I. Dobson & Newman, D. E. *Dynamical and probabilistic approaches to the study of blackout vulnerability of the power transmission grid* in *Proceedings of the 37th Annual Hawaii International Conference on System Sciences* (Institute of Electrical and Electronics Engineers, New York, 2004), 7–14.

192. Dorogovtsev, S. N., Goltsev, A. V. & Mendes, J. F. F. Ising model on networks with an arbitrary distribution of connections. *Physical Review E* **66,** 016104 (2002).

193. Jordan, M. I. *Learning in Graphical Models* (Kluwer Academic Publishers, Dordrecht, 1998).

194. D. Koller & N. Friedman. *Probabilistic Graphical Models: Principles and Techniques* (MIT Press, Cambridge, MA, 2009).

195. Koutra, D. *et al. Unifying guilt-by-association approaches: Theorems and fast algorithms* in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (Springer, 2011), 245–260.

196. Günnemann, W. G. S., Koutra, D. & Faloutsos, C. Linearized and single-pass belief propagation. *Proceedings of the VLDB Endowment* **8** (2015).

197. Bethe, H. A. Statistical theory of superlattices. *Proc. R. Soc. London A* **150,** 552–575 (1935).

198. Mézard, M. & Montanari, A. *Information, Physics, and Computation* (Oxford University Press, Oxford, 2009).

199. Watts, D. J. & Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature* **393,** 440–442 (1998).

200. Eckstein, M., Kollar, M., Byczuk, K. & Vollhardt, D. Hopping on the Bethe lattice: Exact results for densities of states and dynamical mean-field theory. *Physical Review B* **71,** 235119 (2005).

201. Metz, F. L., Neri, I. & Bollé, D. Spectra of sparse regular graphs with loops. *Physical Review E* **84,** 055101 (2011).

202. Bollé, D., Metz, F. L. & Neri, I. *On the spectra of large sparse graphs with cycles* Preprint arxiv:1206.1512 (2012).

203. Newman, M. E. J. Random graphs with clustering. *Physical Review Letters* **103,** 058701 (2009).

204. Yoon, S., Goltsev, A. V., Dorogovtsev, S. N. & Mendes, J. F. F. Belief-propagation algorithm and the Ising model on networks with arbitrary distributions of motifs. *Physical Review E* **84,** 041144 (2011).

205. A. Montanari & T. Rizzo. How to compute loop corrections to the Bethe approximation. *Journal of Statistical Mechanics: Theory and Experiment* **2005,** 10011 (2005).

206. M. Chertkov & Chernyak, V. Y. Loop calculus in statistical physics and information science. *Physical Review E* **73,** 065102 (2006).

207. Newman, M. E. J. & Park, J. Why social networks are different from other types of networks. *Physical Review E* **68,** 036122 (2003).

208. Yedidia, J. S., Freeman, W. T. & Y. Weiss. *Generalized belief propagation* in *Proceedings of the 13th International Conference on Neural Information Processing Systems* (MIT Press, Cambridge, MA, 2000), 668–674.

209. Yedidia, J. S., Freeman, W. T. & Y. Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory* **51,** 2282–2313 (2005).

210. A. Pelizzola. Cluster variation method in statistical physics and probabilistic graphical models. *Journal of Physics A* **38,** R309 (2005).

211. R. Kikuchi. A theory of cooperative phenomena. *Physical Review* **81,** 988–1003 (1951).

212. Yedidia, J. S., Freeman, W. T. & Weiss, Y. in *Exploring Artificial Intelligence in the New Millennium* (eds Lakemeyer, G. & Nebel, B.) 239–270 (Morgan Kaufmann, San Francisco, CA, 2003).

213. Kappen, H. J. & W. Wiegerinck. *Novel iteration schemes for the cluster variation method* in *Advances in Neural Information Processing Systems* (MIT Press, Cambridge, MA, 2002), 415–422.

214. P. Pakzad & V. Anantharam. *Minimal graphical representation of Kikuchi regions* in *Proceedings of the Annual Allerton Conference on Communication Control and Computing* (University of Illinois, Champaign, IL, 2002), 1586–1595.

215. M. Welling. *On the choice of regions for generalized belief propagation* in *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence* (AUAI Press, Banff, 2004), 585–592.

216. Cantwell, G. T. & Newman, M. E. J. Message passing on networks with loops. *Proceedings of the National Academy of Sciences* **116,** 23398–23403 (2019).

217. Kermack, W. O. & McKendrick, A. G. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society A* **115,** 700–721 (1927).

218. R. DurrettR. Durrett. *Spatial Epidemic Models: Their structure and Relation to Data* (Cambridge University Press, Cambridge, 1995).

219. D. Stauffer & S. Solomon. Ising, Schelling and self-organising segregation. *European Physical Journal B* **57,** 473–479 (2007).

220. S. Geman & C. Graffigne. *Markov random field image models and their applications to computer vision* in *Proceedings of the International Congress of Mathematicians* (International Congress of Mathematicians, Berkeley, 1986), 2.

221. M. Yasuda & T. Horiguchi. Triangular approximation for Ising model and its application to Boltzmann machine. *Physica A* **368,** 83–95 (2006).

222. Decelle, A., Krzakala, F., Moore, C. & Zdeborová, L. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E* **84,** 066106 (2011).

223. Zhou, W.-X. & D. Sornette. Self-organizing Ising model of financial markets. *European Physical Journal B* **55,** 175–181 (2007).

224. S. Galam. Rational group decision making: A random field Ising model at T = 0. *Physica A* **238,** 66–80 (1997).

225. D. Stauffer. Social applications of two-dimensional Ising models. *American Journal of Physics* **76,** 470 (2008).

226. MacKay, D. J. *Information Theory, Inference and Learning Algorithms* (Cambridge University Press, Cambridge, 2003).

227. Migon, H. S., D. Gamerman & F. Louzada. *Statistical Inference: An Integrated Approach* (CRC Press, Boca Raton, 2014).

228. N. Friel & J. Wyse. Estimating the evidence—a review. *Statistica Neerlandica* **66,** 288–308 (2012).

229. S. Salinas. *Introduction to Statistical Physics* (Springer, New York, 2001).

230. Baxter, R. J. *Exactly Solved Models in Statistical Mechanics* (Academic Press, London, 1982).

231. Shannon, C. E. Prediction and entropy of printed English. *Bell System Technical Journal* **30,** 50–64 (1951).

232. W. Bialek. *Biophysics: Searching for Principles* (Princeton University Press, Princeton, NJ, 2012).

233. P. Cabral, G. Augusto, M. Tewolde & Y. Araya. Entropy in urban systems. *Entropy* **15,** 5223–5236 (2013).

234. J. Mooij & H. Kappen. On the properties of the Bethe approximation and loopy belief propagation on binary networks. *Journal of Statistical Mechanics: Theory and Experiment* **2005,** P11012 (2005).

235. Krzakala, F. *et al.* Spectral redemption in clustering sparse networks. *Proceedings of the National Academy of Sciences* **110,** 20935–20940 (2013).

236. Martin, T., Zhang, X. & Newman, M. E. J. Localization and centrality in networks. *Physical Review E* **90,** 052808 (2014).

237. Karrer, B., Newman, M. E. J. & Zdeborová, L. Percolation on sparse networks. *Physical Review Letters* **113,** 208702 (2014).

238. A. Allard & L. Hébert-Dufresne. *On the accuracy of message-passing approaches to percolation in complex networks* Preprint arXiv:1906.10377 (2019).

239. F. Mancini & A. Naddeo. Equations-of-motion approach to the spin-1/2 Ising model on the Bethe lattice. *Physical Review E* **74,** 061108 (2006).

240. Dorogovtsev, S. N., Goltsev, A. V. & Mendes, J. F. Critical phenomena in complex networks. *Reviews of Modern Physics* **80,** 1275–1335 (2008).

241. Wolff, U. Collective Monte Carlo updating for spin systems. *Physical Review Letters* **62,** 361–364 (1989).

242. Davis, T. A. & Y. Hu. The University of Florida sparse matrix collection. *ACM Transactions on Mathematical Software (TOMS)* **38,** 1 (2011).

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

244. Newman, M. E. J. Fast algorithm for detecting community structure in networks. *Physical Review E* **69,** 066133 (2004).

245. Rosvall, M. & Bergstrom, C. T. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences* **105,** 1118–1123 (2008).

246. Peixoto, T. P. *Bayesian stochastic blockmodeling* in *Advances in Network Clustering and Blockmodeling* (eds Doreian, P., Batagelj, V. & Ferligoj, A.) (Wiley, New York, 2019), 289–332.

247. Fortunato, S. Community detection in graphs. *Physics Reports* **486,** 75–174 (2010).

248. Guimerà, R., Sales-Pardo, M. & Amaral, L. A. N. Modularity from fluctuations in random graphs and complex networks. *Physical Review E* **70,** 025101 (2004).

249. Massen, C. P. & Doye, J. P. K. Identifying "communities" within energy landscapes. *Physical Review E* **71,** 046101 (2005).

250. Reichardt, J. & Bornholdt, S. Statistical mechanics of community detection. *Physical Review E* **74,** 016110 (2006).

251. Good, B. H., de Montjoye, Y.-A. & Clauset, A. Performance of modularity maximization in practical contexts. *Physical Review E* **81,** 046106 (2010).

252. Riolo, M. A. & Newman, M. E. J. Consistency of community structure in complex networks. *Physical Review E* **101,** 052306 (2020).

253. Peixoto, T. P. Revealing consensus and dissensus between network partitions. *Physical Review X* **11,** 021003 (2021).

254. A. Lancichinetti & S. Fortunato. Consensus clustering in complex networks. *Scientific Reports* **2,** 1–7 (2012).

255. J. Calatayud, R. Bernardo-Madrid, M. Neuman, A. Rojas & M. Rosvall. Exploring the solution landscape enables more reliable network community detection. *Physical Review E* **100,** 052308 (2019).

256. Vinh, N. X., Epps, J. & Bailey, J. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for Chance. *Journal of Machine Learning Research* **11,** 2837–2854 (2010).

257. Grünwald, P. D. & A. Grünwald. *The Minimum Description Length Principle* (MIT Press, Cambridge, MA, 2007).

258. J. Tabor & P. Spurek. Cross-entropy clustering. *Pattern Recognition* **47,** 3046–3059 (2014).

259. Wallace, R. S. & T. Kanade. *Finding natural clusters having minimum description length* in *10th International Conference on Pattern Recognition* (Institute of Electrical and Electronics Engineers, Hoboken, 1990), 438–442.

260. T. Li, S. Ma & M. Ogihara. *Entropy-based criterion in categorical clustering* in *Proceedings of the Twenty-first International Conference on Machine Learning* (Association for Computing Machinery, New York, 2004), 68.

261. M. Narasimhan, N. Jojic & Bilmes, J. A. Q-clustering. *Advances in Neural Information Processing Systems* **18,** 979–986 (2005).

262. O. Georgieva, K. Tschumitschew & F. Klawonn. *Cluster validity measures based on the minimum description length principle* in *Proceedings of the International Conference on Knowledge-Based and Intelligent Information and Engineering Systems* (Springer-Verlag, Berlin, 2011), 82–89.

263. Rosvall, M. & Bergstrom, C. T. An information-theoretic framework for resolving community structure in complex networks. *Proceedings of the National Academy of Sciences* **104,** 7327–7331 (2007).

264. Cover, T. M. & Thomas, J. A. *Elements of Information Theory* (John Wiley, New York, 1991).

265. Newman, M. E. J., Cantwell, G. T. & Young, J.-G. Improved mutual information measure for clustering, classification, and community detection. *Physical Review E* **101,** 042304 (2020).

266. Doane, D. P. Aesthetic frequency classifications. *The American Statistician* **30,** 181–183 (1976).

267. P. Hall. Akaike's information criterion and Kullback-Leibler loss for histogram density estimation. *Probability Theory and Related Fields* **85,** 449–467 (1990).

268. Peixoto, T. P. Merge-split Markov chain Monte Carlo for community detection. *Physical Review E* **102,** 012305 (2020).

269. Peixoto, T. P. Nonparametric Bayesian inference of the microcanonical stochastic block model. *Physical Review E* **95,** 012317 (2017).

270. Fortunato, S. & Barthélemy, M. Resolution limit in community detection. *Proceedings of the National Academy of Sciences* **104,** 36–41 (2007).

271. Bearman, P. S., Moody, J. & Stovel, K. Chains of affection: The structure of adolescent romantic and sexual networks. *American Journal of Sociology* **110,** 44–91 (2004).

272. Udry, J. R., Bearman, P. S. & Harris, K. M. *National Longitudinal Study of Adolescent Health* This research uses data from Add Health, a program project directed by Kathleen Mullan Harris and designed by J. Richard Udry, Peter S. Bearman, and Kathleen Mullan Harris at the University of North Carolina at Chapel Hill, and funded by grant P01–HD31921 from the Eunice Kennedy Shriver National Institute of Child Health and Human Development, with cooperative funding from 23 other federal agencies and foundations. Special acknowledgment is due Ronald R. Rindfuss and Barbara Entwisle for assistance in the original design. Information on how to obtain the Add Health data files is available on the Add Health website (http://www.cpc.unc.edu/addhealth). No direct support was received from grant P01–HD31921 for this analysis. 1997.

273. Kirkley, A. Information theoretic network approach to socioeconomic correlations. *Physical Review Research* **2,** 043212 (2020).

274. S. Ferson, V. Kreinovich, J. Hajagos, W. Oberkampf & L. Ginzburg. Experimental uncertainty estimation and statistics for data having interval uncertainty. *Sandia National Laboratories, Report SAND2007-0939* **162** (2007).

275. Von Hippel, P. T., Scarpino, S. V. & I. Holas. Robust estimation of inequality from binned incomes. *Sociological Methodology* **46,** 212–251 (2016).

276. Council, N. R. *Using the American Community Survey: Benefits and Challenges* (National Academies Press, Washington D.C., 2007).

277. Spielman, S. E., D. Folch & N. Nagle. Patterns and causes of uncertainty in the American Community Survey. *Applied Geography* **46,** 147–157 (2014).

278. Feng, S. & Kirkley, A. Mixing patterns in interdisciplinary co-authorship networks at multiple scales. *Scientific Reports* **10,** 1–11 (2020).

279. Chung, F. & Lu, L. The average distances in random graphs with given expected degrees. *Proceedings of the National Academy of Sciences* **99,** 15879–15882 (2002).

280. E. Schneidman, Berry II, M. J., R. Segev & W. Bialek. Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature* **440,** 1007–1012 (2006).

281. F. Morcos *et al.* Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences* **108,** 1293–1301 (2011).

282. T. Bury. Market structure explained by pairwise interactions. *Physica A* **392,** 1375–1385 (2013).

283. Nguyen, H. C., R. Zecchina & J. Berg. Inverse statistical problems: From the inverse Ising problem to data science. *Advances in Physics* **66,** 197–261 (2017).

284. A. Shamir. *A survey on mesh segmentation techniques* in *Computer Graphics Forum* **27** (The Eurographics Association and John Wiley & Sons, 2008), 1539–1556.

285. Lusher, D., Koskinen, J. & Robins, G. *Exponential Random Graph Models for Social Networks: Theory, Methods, and Applications* (Cambridge University Press, Cambridge, 2012).

286. Newman, M. E. J. & Barkema, G. T. *Monte Carlo Methods in Statistical Physics* (Oxford University Press, Oxford, 1999).

287. Dempster, A. P., Laird, N. M. & Rubin, D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B* **39,** 185–197 (1977).

288. Decelle, A., Krzakala, F., Moore, C. & Zdeborová, L. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E* **84,** 066106 (2011).

289. Newman, M. E. J. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E* **74,** 036104 (2006).

290. B. Hayes. Connecting the dots: Can the tools of graph theory and social-network studies unravel the next big plot? *American Scientist* **94,** 400–404 (2006).