6 Interlinkages and Heterogeneity: Network Theory

6.1 Introduction

One of the key signatures of a complex system is the emergence of patterns which cannot be understood by isolating the constituent parts of the system. It is the interaction that leads to emergence of patterns. Network theory has become an integral part of studying complex systems due to the simplicity of describing very complicated nature of connections through a standardized mathematical object. At the same time, theory of networks made its inroad into the data science literature due to its algebraic simplicity and manipulability. As we will show, matrix algebra can take us far into studying the network structure of a large variety of interlinkages that can be found in the physical, social and economic worlds. The widespread applicability of the network theory along with the ability to compress information, has made network theory a suitable candidate to tie data science with complex systems. In the following, we first provide an intuitive picture of exactly what are networks? Then we will proceed towards describing the mathematics behind it.

In an abstract sense, networks are collections of a set of nodes and edges connecting them. The definition is so general that it would be difficult to think of a somewhat complex object that cannot be expressed as a network. As some of the following examples will show, an astounding number of things we see around us are essentially networks. Some of them are more tangible than others. Let us consider a fw of them in order to develop an intuitive understanding before getting into the mathematical details. For example, connectivity through public or private transport gives rise to transportation network (Banavar et al., 1999), and the roads connecting different places give rise to road network (Zhan and Noon, 1998). While making phone calls, we use the mobile phone communication network (Onnela et al., 2007; Deville et al., 2014). Communications through emails gives rise to email networks (Newman et al., 2002a). In the above three example, the road network is clearly more tangible than the other two. Building on the similar line of thought, some other examples would include railways network (Sen et al., 2003a), internet (Amaral et al., 2000; Yook et al., 2002), power grid network (Albert et al., 2004; Arianos et al., 2009; Motter et al., 2013; Pagani and Aiello, 2013) and World Wide Web (Huberman, 2001). All of these fall under the rubric of socio-technical networks.

We can enhance the scope of examples a bit further to understand actions and

strategies of human beings. The concept of social networks are widespread and immensely useful to understand the nature of interactions (Newman and Park, 2003), which represents a group of people connected together through certain events or linkages. Much of the human social behavior can be traced back to the linkages that human beings create in various forms. Examples range from online communication network such as Facebook (currently, a part of Meta), twitter, etc. (Lewis et al., 2008), sexual contact networks (Liljeros et al., 2003), caste and class networks (Davis et al., 1941; Munshi and Rosenzweig, 2016), scientific collaboration networks (Newman, 2001), migration networks (Nagurney et al., 1992; Munshi, 2003), movie actors networks (Watts and Strogatz, 1998; Herr et al., 2007), terror attack networks (Galam and Mauger, 2003; Clauset et al., 2007) and citation networks (Price, 1965; Hummon and Dereian, 1989) to name a few. Even animals have their social networks (Lusseau, 2003; Farine and Whitehead, 2015).

Biological and physical networks are also abound, viz. protein-protein interaction networks (Jeong et al., 2001), brain networks (Fornito et al., 2016; Bullmore and Sporns, 2012), nervous system network (Pan et al., 2010), metabolic network (Jeong et al., 2000), food web network (Dunne et al., 2002), disease transmission networks (Riley, 2007; Valdez et al., 2018), earthquake network (Abe and Suzuki, 2004) and river network (Maritan et al., 1996; Dodds and Rothman, 2000) to name a few.

These networks have been analyzed in great details in the references provided above. For our purpose, we will stick to economic and social networks. It is important to note that while networks have a common underlying structure, they differ a lot in terms of which system they belong to. In the following, we will almost exclusively analyze economic and social networks with appropriate interpretations and references. For more details on technical, physical and biological networks, one can refer to Newman (2010) and Caldarelli (2007) as a starting point.

Historically, the study of networks started from graph theory. One of the earliest problems was the famous *Seven Bridges of Königsberg* problem that in some sense gave birth to the modern graph theory. Königsberg is a historic Prussian city (it was earlier a part of Germany and now it is in Russia) had 7 bridges over a river connecting four points. The problem was to find a path to cross each of seven bridges in the city only once without crossing any bridge twice (figure 6.1). Almost three centuries back, Leonhard Euler showed that this problem does not have a solution. More importantly, his formulation provided one of the first formal treatments of graphs as a mathematical object (Euler, 1736). A major conceptual leap his analysis provided was to recognize that the underlying physical locations or distances do not matter. In other words, one can move away from the physical world and consider an abstract version of the problem only in terms of a network with four nodes and seven links. This separation provided the conceptual impetus to study the network as a separate mathematical object.

In the recent times, advent of social media has made the word 'network' a commonplace word. The present day popularity of for social networks (including a blockbuster Hollywood movie with the name *The Social Network* based on Mark Zuckerberg who is currently the CEO of Meta which has subsumed Facebook)

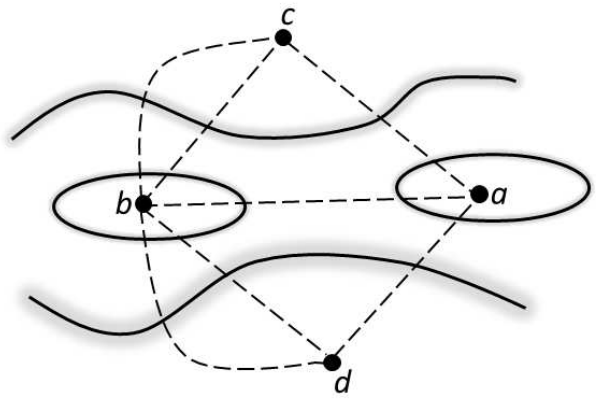

Figure 6.1 Layout of the Königsberg bridge problem. Points a and b are in two islands, points c and d are in the mainland with a river flowing in between. Arcs and the lines connecting the four points (a, b, c and d), show possible paths to go from one point to another via bridges.

can be traced back to a pioneering analysis by Jacov Moreno, a psychiatrist. He produced one of the first systematic studies on social networks, which he found by mapping the friendships between girls in a reform school in New York city (Moreno, 1953).

While the diversity of different types of networks that we see around us is astounding, a follow up question appears that if networks are so general, is there any unifying principle or methodology that unites or finds some deep patterns in these different types of networks? Or is it simply a descriptive device? The answer as we will argue in the rest of the chapter is that it allows us to do both. In one form, networks are indeed descriptive devices that allow us to summarize information of a set of interconnected entities. The abstraction becomes very useful to map very different systems into the same mathematical object. However, the theory of networks have been also very successful in not only in terms of description, but also in providing analytical methods to investigate causal phenomena.

Before delving into the algebraic description of networks, it is useful to take a detour through the concept of hierarchy vis-á-vis network. Simply put, hierarchies are vertically organized structures whereas networks are more horizontally organized structures. we can borrow some insights from historian Niall Ferguson to understand why networks matter beyond the simple description of linkages in the socio-economic context. In his book – *The Square and the Tower* (Ferguson, 2017), Ferguson argues that the current dominance of the 'networked era' in public

thinking is shaped by the social media revolution, which is often presented as an unprecedented phenomenon. But an earlier networked era probably followed from the invention of printing press in Europe. These two era have been separated by an intermediate period spanning hundreds of years that saw the rise and collapse of hierarchies in the organization of the society. Following this idea, it is tempting to conjecture that many of the socio-economic networks that we see around us also saw a corresponding dynamics in their evolution. For a very specific example of how technological networks are related to economic and political networks, one can refer to Wenzlhuemer (2013) who discusses how the advent of telegraph network contributed to globalization in the nineteenth century.

Below we will review and discuss some basic statistics and algorithms to describe networks in a more general context, going beyond the socio-economic world. Later on, in chapter 7, we will come back to the discussions around social and economic competitions, and how linkages may or may not affect the emergent properties.

6.2 Parts of a network

A set of nodes (also called vertices) connected to each other through some relationship among them (called edges or links) is known as a network (graph). A network can be directed or undirected and weighted or unweighted (see figure 6.2). In real world networks, nodes can be anything like people, cities, stocks, etc. and edges can be friendship, distance, similarity, etc. Notationally, we will describe a network of N nodes as $\mathcal{N} = \{V,\ E\}$ where V denotes a set of nodes (with cardinality N) and E denotes the corresponding set of edges. We can define two types of networks based on the directions of the edges and the weights of the edges.

It is customary to describe and/or analyze a network through its adjacency matrix. Mathematically, the adjacency matrix $\mathcal{A}$ is an $N \times N$ matrix where the i, j-th element denotes the relationship between the i-th node and the j-th node, for all $i, j \in V$.

6.2.1 Undirected and directed networks with edge weights

As the name suggests, edges in undirected networks do not have directions. An example of such connection is Facebook friendship. If two people are friends on Facebook, then the *edge* exists both ways. As opposed to say Twitter followers. When a person follows another person in Twitter, it is not mandatory for the person being followed to follow the follower. Thus the relationship can be asymmetric. This type of networks are called directed networks.

It is quite easy to describe directed and undirected networks in terms of the adjacency matrices. But before that, it is useful to describe the weight of an edge. The elements in the adjacency matrix $\mathcal{A}_{ij}$ represents the weight of the edge connecting node i to j. Weight represents a measure of the importance of an edge or a link in a

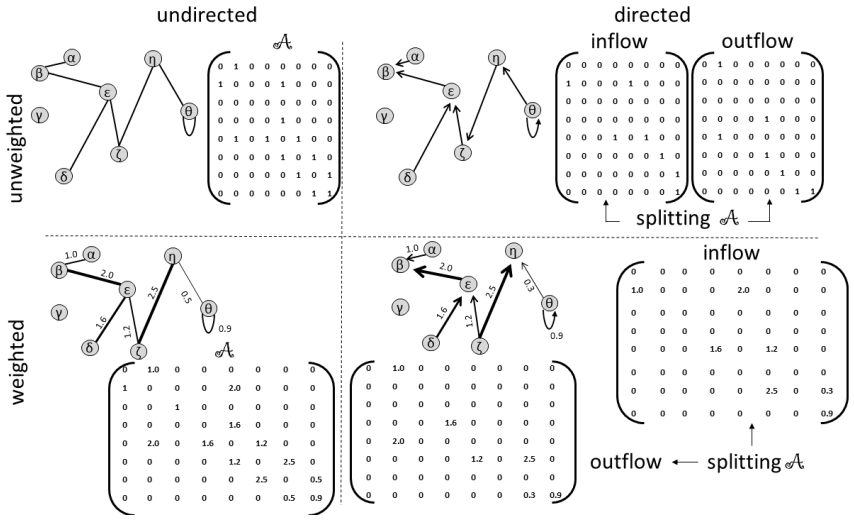

Figure 6.2 Graphical representation of networks (undirected and directed with and without weights) along with their corresponding adjacency matrices, $\mathcal{A}$. The circles represent nodes and the lines (undirected) or arrows (directed) connecting those circles represent links or edges. The widths of the links represent the strengths or weights of the links.

network. When all existing edges has the same weight in the network, the network is called *unweighted*. A weighted network, on the other hand, will display variation in weights across edges. A standard way to describe an unweighted network is use binary weights such that if an edge exists between the nodes i and j (the link beginning from node j and ending at node i – we will describe the notation more elaborately below), we write $\mathcal{A}_{ij} = 1$, else $\mathcal{A}_{ij} = 0$.

If for all pairs of nodes $\{i,\ j\}$, the values in the adjacency matrix are symmetric for i to j and j to i, then the network is undirected. Otherwise, it is directed. To put it differently, if there exists at least one pair of nodes $\{i,\ j\}$ such that $\mathcal{A}_{ij} \neq \mathcal{A}_{ji}$, then the network if directed. Edges in directed networks can be differentiated in terms of incoming and outgoing edges. Examples of real-life networks which are directed are email networks, transportation networks, asset-holding networks and so forth.

We show some examples of weighted and unweighted networks along with directions in figure 6.2. There is one node in each of these panels which is unconnected to the rest of the network. There are four combinations shown in four panels, unweighted-undirected, unweighted-directed, weighted-undirected and weighted-directed. Each panel shows the plots of an example network and the corresponding adjacency matrix. Directed edges are indicated by arrows and different edge weights are indicated by varying edge widths. In table 6.1, we provide a few examples of real-life networks along with the nature of the nodes, links and direction.

Throughout this chapter, we will consistently refer to two different types of net-

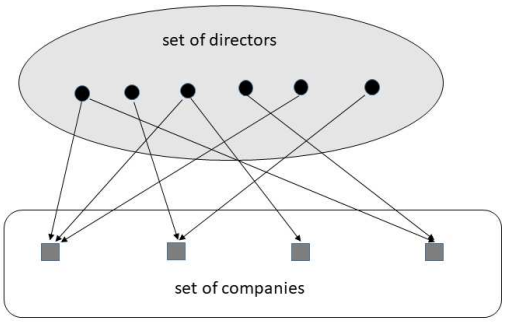

Figure 6.3 Corporate network: A schematic illustration of a bipartite network capturing director-to-company mapping.

works to develop the ideas. The first one is a social network in the form of board interlocks and the second one is an economic network in the form of production linkages in a given country. We will explain them in details going forward.

Bipartite networks and projections

In order to explain the ideas that we are going to describe, we will depend on a particular type of social network. This is a specific *bipartite* network of corporate board interlock via directors (Aggarwal et al., 2020).

We will describe what is a bipartite network below. Before that, let us see an example to form an idea about the object. In figure 6.3 we show a schematic of a network comprising two sets of nodes – a set of company directors and a set of companies which they are parts of. This board network is called a *bipartite* network of directors and boards.

This bipartite network leads to two different networks. It can be projected only on the companies where the nodes will be companies and a pair of companies will be connected if they share a common director, or it can also be projected only on the directors where a pair of directors will be connected if they are present in the board of the same company. We will refer to the first one as the board interlock network and the second one as the directors' network. In the following we use data sampled from the Prime database for describing this network.

Generally, bipartite networks represent a complex version of interconnections. Specifically, they have two types of nodes and the edges always connect nodes of one type to nodes of another types, with no edge connecting the same type of nodes. Examples of such sets of nodes are movie-actor networks, country-trade, consumer-products etc., where the connections are across the sets of nodes.

The *incidence matrix* $\mathcal{I}$ describe the relationship among two types of entities. They can be nodes and edges. In this case, we can utilize them to denotes two sets

Table 6.1 Illustration of some real-life networks with nodes, links and direction

Network	Nodes	Links	Direction	Reference (among many others)
Scientific Collaboration	Scientists	Co-authorship	Undirected	Newman (2004a)
Social Media	People	Following/ friendship	(Un-)directed	Lee et al. (2014)
Email	Email id	Emails	Directed	Newman et al. (2002a)
Transportation	Locations	Vehicles	Directed	Sen et al. (2003b)
Politics	People or Organizations	Relations	Directed	Gil-Mendieta and Schmidt (1996)
Migration	People	Movement	Directed	Munshi (2003)
Finance	Banks and firms	Assets	Directed	Acemoglu et al. (2015)
Markets	Buyers and Sellers	Trading	Directed	Choi et al. (2017)
Innovation	Patents	Citation	Directed	Acemoglu et al. (2016)

of nodes. If there are M nodes of type one and N nodes of type two, then $\mathcal{I}$ is an $M \times N$ rectangular matrix. We can represent an unweighted bipartite network by an incidence matrix $\mathcal{I}$ such that if there exists a link between node i of type one and node j of type two, then $\mathcal{I}_{ij} = 1$, else 0. If each node in one group connects to all nodes in the other group, then we will call such networks *complete* bipartite networks.

6.3 Node- and network-level characteristics

In this section, we will describe some fundamental network measures. These measure are generally applicable to all networks and hence, provides an useful set of tools and techniques to summarize global and local properties of networks.

6.3.1 Degree

Degree indicates popularity or importance of a node in a network. Consider an undirected, unweighted network with adjacency matrix $\mathcal{A}$. The degree of node $i \in V$

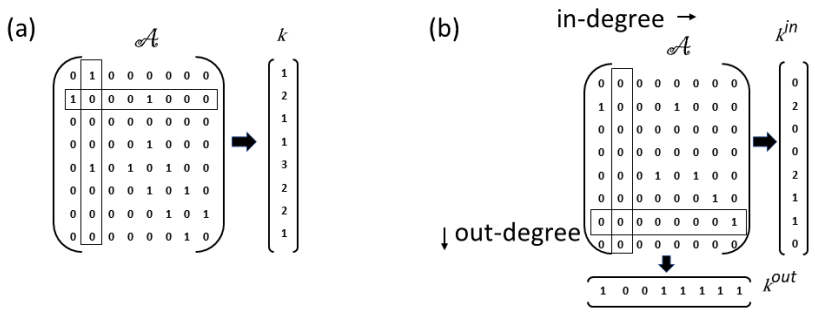

Figure 6.4 Schematic representation of in-degree, out-degree, and total degree of the unweighted networks shown in Figure 6.2 excluding the self-loop. We illustrate the connectivity of nodes through adjacency matrix $\mathcal{A}$ for undirected and directed networks, (a) total degree across nodes in the undirected network. (b) in-degree and out-degree across nodes in the directed network. We have excluded the self-loop from our calculation to capture only the linkages across nodes.

is the sum of number of links a node i has (see figure 6.4, panel (a)). Notationally, we write degree of the i-th node as

$$k_i = \sum_{j=1}^{N} \mathcal{A}_{ij} = \sum_{j=1}^{N} \mathcal{A}_{ji}. \tag{6.1}$$

The *average* degree of a network is defined as

$$\langle k \rangle = \frac{\sum_i^N k_i}{N} \tag{6.2}$$

where $\langle . \rangle$ denotes sample average. Note that the maximum possible degree of a node is $k_{max} = N - 1$, obtained when the node is connected to every other node. We define the density of connections as the ratio between the average and maximum degree.

As we have seen in section 6.2.1, directed networks can potentially have both incoming and outgoing links. The number of incoming links of node i is called *in-degree*. It is important to define the adjacency matrix in a clearer way for this purpose. Technically speaking, we can use either $\mathcal{A}_{ij}$ or $\mathcal{A}_{ji}$ to denote an outgoing link from the node i to j. In the present context, we utilize the notation $\mathcal{A}_{ij}$ to denote an outgoing link from node j to i – equivalently, that denotes an incoming link to node i from node j.

With such notation, the in-degree of the i-th node is sum of row entries in adjacency matrix $\mathcal{A}$ (see figure 6.4, panel(b)) i.e.,

$$k_i^{in} = \sum_{j=1}^{N} \mathcal{A}_{ij}. \tag{6.3}$$

The number of outgoing links (for a binary network) of node i is called *out-degree* of that node. More generally, the out-degree is sum of the column entries in adjacency matrix $\mathcal{A}$ i.e., for the i-th node, the out-degree is

$$k_i^{out} = \sum_{j=1}^{N} \mathcal{A}_{ji}. \tag{6.4}$$

We can extend the concept of in-degree and out-degree in case of a weighted network as well. The sum of the weights of its links defines the *weighted degree*. Therefore, the weighted degree of node i in an *undirected weighted* network is defined as

$$k_i^{w} = \sum_{j=1}^{N} w_{ji} = \sum_{j=1}^{N} w_{ij} \tag{6.5}$$

where w_{ji} represents the weight of the link between nodes i and j and $w_{ji} = 0$ if no link is present from node j to i. By the same logic, for directed weighted networks we can write the weighted in-degree and out-degree of the i-th node as

$$k_i^{in,w} = \sum_{j=1}^{N} w_{ij} \quad \text{and} \quad k_i^{out,w} = \sum_{j=1}^{N} w_{ji}. \tag{6.6}$$

6.3.2 Dense and sparse networks

Density is a measure of how dense the connections are in a given network. There are two extreme examples. A complete network is a network for which all nodes are connected to every other node. On the other hand, a ring network is where each node has exactly two neighbors such that the whole network creates a closed loop. Intuitively, a complete network is clearly more *dense* than a ring network even with the same number of nodes. How do we formally capture this idea of density? A technical way to answer this question would be to count how many links are present in the network as a fraction of the total possible links.

Intuitively, such a measure would range from 0 (no connectivity) to 1 (fully connected) after normalization. In real-world networks, the density is often much smaller than 1. The reason is that a large fraction of pairs of nodes may not have direct connections between themselves. If the density is very low, such networks are called *sparse* networks. The fewer edges are in a network, the sparser the network is.

The maximum number of links in an undirected, directed and bipartite networks are:

$$L_{max} = \begin{cases} \frac{N(N-1)}{2} & \text{for undirected network with } N \text{ nodes,} \\ N(N-1) & \text{for directed network with } N \text{ nodes.} \end{cases} \tag{6.7}$$

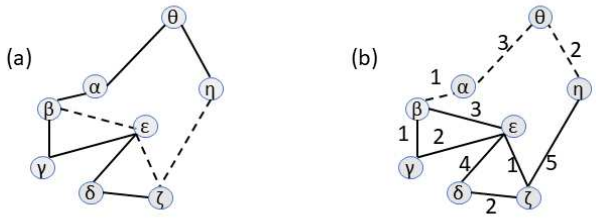

Figure 6.5 Schematic representation of path length of unweighted and weighted network. Shortest-paths (dashed line) in (a) unweighted undirected network, and (b) weighted undirected network. In each case the shortest-path between nodes β and η is highlighted in dashed lines. (a) the shortest-path length between node β and η in unweighted network is $l_{\beta,\eta} = 3$: $\{\beta, \epsilon, \xi, \eta\}$, (b) the shortest-path length in weighted network is $l_{\alpha,\eta} = 3 : \{\beta, \alpha, \theta, \eta\}$.

Therefore, the density of a network with N nodes and L links is

$$d = L/L_{max} = \begin{cases} \frac{2L}{N(N-1)}, & \text{for undirected network with } N \text{ nodes,} \\ \frac{L}{N(N-1)}, & \text{for directed network with } N \text{ nodes.} \end{cases} \tag{6.8}$$

For a *growing* network, the growth can be observed in terms of how the number of links increases as a function of the number of nodes. If the number of links grows proportional to the number of nodes ($L \sim N$), then the network is sparse. If it grows faster than linear (e.g. quadratically $L \sim N^2$) or exponentially ($L \sim e^N$) with respect to the network size, then the network is called dense.

6.3.3 Paths and distances across nodes

A path from a given *source* node to a given *target* node is a sequence of edges traversed to go from the *source* node to the *target* node. For an unweighted and undirected network, the number of edges traversed in a path is called the *path length*. A given pair of nodes may have multiple paths. These paths may have different lengths, and may or may not share some common edges. If the source and the target nodes are the same, the path is called *cyclic*. If a path never goes through the same link more than once, we call it *simple*.

The concept of a path is the basis of the definition of *distance* among nodes in a network. A straight-forward way to conceptualize distance between a pair of nodes is to consider the minimum number of edges to be traversed in a path connecting

the two nodes. In other words, distance is simply captured by the *shortest path*, and its magnitude is given by the *shortest-path length*. There may be multiple shortest paths between two nodes, but they all must have the same length. The shortest paths between two nodes depend on the type of network, i.e. whether the network is directed and/or weighted or not. In the case of an undirected and unweighted network, the shortest path is just the one that has the minimum number of edges. In case of directed network, the paths must be consistent with the direction of the links along the path.

The *average shortest-path length* is obtained by averaging the shortest-path lengths across all pairs of nodes. The *diameter* of the network is the maximum shortest-path length across all pairs of nodes, i.e. the length of the longest shortest path in the network.

Mathematically, the average path length of an undirected or directed, unweighted or weighted network is defined as

$$\langle l \rangle = \begin{cases} \frac{2\sum_{i,j}^{N} d_{ij}}{N(N-1)} & \text{for an undirected network,} \\ \frac{\sum_{i,j}^{N} d_{ij}}{N(N-1)} & \text{for a directed network.} \end{cases} \tag{6.9}$$

where d_{ij} is the shortest-path length between nodes i and j, and N is the number of nodes. Using the same notation, we can write the diameter as

$$dim = \max_{i,j \in V} \quad d_{ij}. \tag{6.10}$$

In figure 6.5, we show two networks – one is unweighted and the other one is weighted. In both, the minimum path between nodes β and η are highlighted and drawn by dashed lines.

6.3.4 Connectedness of nodes

For many processes on networks like information diffusion (think of a rumor), the network topology is very important. One key feature of such topological properties is *connectedness*. How far information diffuses in a given social network would depend on how fragmented is the network. With a highly fragmented network, information cannot jump from one component of the network to another. However, if the network is connected in the sense that from any node one can find a path to any other node, chances are higher that the information diffusion can actually take place.

Thus, connectedness is an important property to understand the structure and function of a network. Note that the number of edges in a network is bound by the number of nodes (we are ignoring the possibility of having multiple edges connecting the same pair of nodes). The upper bound corresponds to a complete network whereas in the lower bound, a network would have no links at all. In section 6.3.2, we saw that the higher the density, the greater are the chances that the network is *connected*. The fewer the links and the lower the density, the higher the chances that the network is disconnected, which gives rise to smaller connected components in the network.

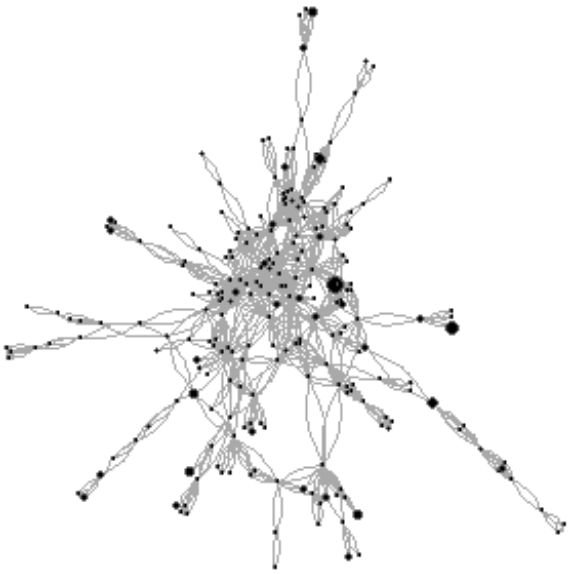

Figure 6.6 Corporate network: An illustration of the giant component of Indian corporate network of a group of randomly sampled 600 directors in the year 2016. Nodes represents the directors of Indian company registered in the National Stock Exchange and a link is formed when two directors are connected, i.e., if they belong to the board of the same company. We do not show single nodes.

A component is a connected subnetwork which is unconnected to other subnetworks. The largest connected components in many real-world networks include a substantial portion of the network and is often called the *giant* component. In a connected network, the giant component coincides with the entire graph.

An undirected network can either be connected or unconnected. A directed network on the other hand can be *strongly* connected, *weakly* connected or unconnected. In a strongly connected network, there is at least one directed path between every pair of nodes, in both directions. A weakly connected network is such that replacing all the directed edges with undirected edges will create a connected network, which is undirected.

As an example, in figure 6.6 we plot the giant component of the corporate network with a sample of 600 directors of firms registered in the National Stock Exchange in India. Two directors are connected if they belong to the same company's board. Each director can hold multiple director positions. In the figure, the giant com-

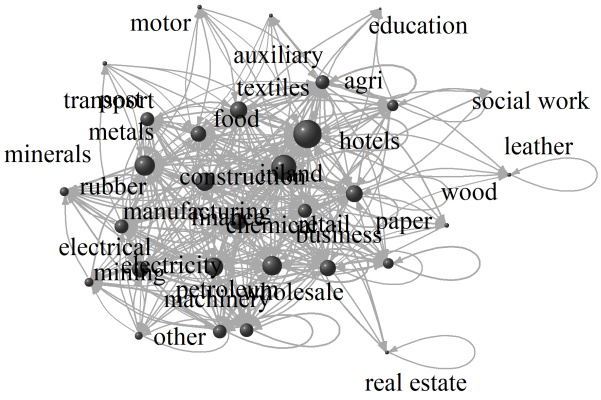

Figure 6.7 Production network: An illustration of the Indian sectoral production network with 33 sectors in 2010. Edge weights have been normalized and thresholded for ease of visualization. Node sizes have been scaled with respect to the weights of the incoming and outgoing links.

ponent has 230 directors out of the total sample of 600 directors considered. we have not plotted the rest of them as there would be too many small subnetworks to visualize. For the same reason, we do not consider the full set of directors in this figure as that would be in the order of many thousands and a clean visualization would not be possible.

We show a different type of empirical network in the form of the sectoral production network in India. The data is obtained from publicly available sector-to-sector flow of goods and services. Figure 6.7 shows a normalized and thresholded version of the network in 2010. This network notably has self-loops. For example, the construction sector may utilize construction services themselves.

6.3.5 Clusters of nodes

Transitivity is the tendency of the adjacent nodes in a network to be connected. Intuitively, it refers to a situation in a digraph in which node i is connected to node j, node j is connected to node k, and node i is connected to node k. In other words, the idea is built on the popular adage that *friends of my friends are my friends*. From the description, it is evident that transitivity depends on the existence of triads i.e., subgraphs formed by three nodes.

A quantitative way to capture the measure would be to consider the clustering coefficient. One can also think of clustering in the form of the likelihood of finding triplets. Consider an undirected network $\mathcal{A}$. If $\mathcal{A}_{ij} = 1$ and $\mathcal{A}_{ik} = 1$ then the nodes

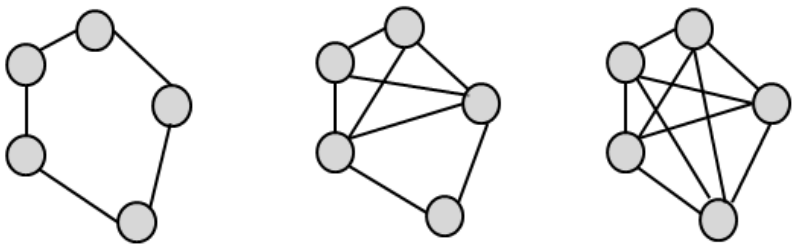

Figure 6.8 Schematic representation of clustering coefficient. Illustration of the global clustering coefficient of three networks: $C = 0$, $15/19$ and 1 respectively. The dashed lines represent missing links among the nodes.

i, j, k form a triplet. Triplets can be of two types: triangular and non-triangular. Triangles contain 6 paths of length 3 whereas non-triangles contain two paths of length 2. Based on the above values of $\mathcal{A}_{ij}$ and $\mathcal{A}_{ik}$, $\mathcal{A}_{jk}$ would be equal to 1 for a triangle whereas $\mathcal{A}_{jk}$ would be equal to 0 for a non-triangle (or open-triplet).

Empirical studies have revealed that often for social networks, most of the triplets are actually triangles. To give an example, if Alagie is friends with Bikul and Chan, then it is quite likely that Bikul and Chan are also direct friends of each other. This simple observation led to the formulation of a simple measurement of transitivity of an undirected unweighted network. It can be measured simply by the fraction of all possible triplets which are triangles –

$$C = \frac{3 \times \#triangles}{\#connected\ triplets}. \tag{6.11}$$

Formally, it is called the global *clustering coefficient*. Although there were precursors to this measure, probably Newman et al. (2002b) made this measure popular in network analysis. The value of C lies between 0 and 1 (figure 6.8). The measure C takes the value of 1 when all triplets in the network are triangles, and C takes the value of 0 when no triangle exists.

Watts and Strogatz (1998) introduced a measure of clustering coefficient at the node level as the ratio of existing edges with respect to maximum number of edges, and at the network level, by simply averaging over all nodes. Consider a node i with degree k_i. Clearly, the maximum possible number of connections among the

neighbors is

$$n_i^{max} = \frac{k_i(k_i - 1)}{2}. \tag{6.12}$$

The local clustering coefficient of node i is defined as

$$C(i) = \frac{2n_i}{k_i(k_i - 1)} \tag{6.13}$$

where n_i is the actual number of connections among the neighbors of i. Note that $C(i)$ is only defined if the degree k_i is greater than 1 (due to the $k_i - 1$ in the denominator). Intuitively, a node must have at least two neighbors for any triangle to be possible. One can construct an analogous measure for directed network in a similar form:

$$C(i) = \frac{n_i}{k_i(k_i - 1)}. \tag{6.14}$$

Here the factor of 2 is removed since a connection A_{ij} is different from A_{ji} and therefore, they are counted separately. We can define the clustering coefficient of a network as a whole, as the mean of the clustering coefficients across all of its nodes –

$$\langle C \rangle = \frac{1}{N} \sum_{i=1}^{N} C_i \tag{6.15}$$

where N is the number of nodes in network. Note that this is different from the global clustering coefficient given by equation 6.11. A low clustering coefficient (near zero) means that the network has few triangles, while a high clustering coefficient (near one) means that the network has possibly many triangles.

As noted above, social networks typically display high degree of clustering. Another term that is often used to describe the same phenomenon is "*triadic closure*". Online social networks make suggestions based on these kind of triadic closures. For example, social media platforms recommend "people you may know" based on common friends, where as there are social media which recommends accounts followed by one's friends. For the second type of social media, we can adapt the adage of friend triplets to the following: "followers of my followers are also my followers"!

6.3.6 Reciprocity of connections

Clustering and transitivity deals with how network triplets behave, which carries information about the local topology of the nodes. However, we can look at more restricted local topology by considering *reciprocity*. The idea is to quantify if a node have a directed link i to j, how likely is it to have a link in the direction j to i? Reciprocity is a quantitative measure used to determine the tendency of node pairs to form mutual connections in directed network. Needless to say, this would not make much sense in an undirected network. The measure R can be defined as the ratio of the number of links going in both direction with respect to the total number

of links. For an adjacency matrix $\mathcal{A}$, we write

$$R = \frac{TrA^2}{||\mathcal{A}||} \tag{6.16}$$

where $0 \leq R \leq 1$. The case $R = 1$ corresponds to a completely bidirectional network and $R = 0$ corresponds to a fully unidirectional network. Note that reciprocal links are effectively self-loops with length 2. The numerator captures this number via squaring the adjacency matrix and taking its trace. The denominator is given by

$$||\mathcal{A}|| = \sum_{ij} \mathcal{A}_{ij}, \tag{6.17}$$

which is clearly the total number of links in a directed network. It can be confirmed that going by this formula, a symmetric directed network has $R = 1$.

Garlaschelli and Loffredo (2004) defined reciprocity as the correlation coefficient between the entries of the adjacency matrix of a directed network $\mathcal{A}$ in the following form:

$$R_{GL} \equiv \frac{\sum_{i \neq j}(\mathcal{A}_{ij} - \overline{\mathcal{A}})(\mathcal{A}_{ji} - \overline{\mathcal{A}})}{\sum_{i \neq j}(\mathcal{A}_{ij} - \overline{\mathcal{A}})^2}. \tag{6.18}$$

The average value $\overline{\mathcal{A}} \equiv \frac{\sum_{i \neq j} \mathcal{A}_{ij}}{N(N-1)}$ in turn is equal to the ratio of the number of edges and $N(N-1)$. The measure R_{GL} being greater than 0 indicates reciprocal graphs, and R_{GL} being lesser than 0 indicates anti-reciprocal networks.

6.3.7 Homophily in a network

So far we have been dealing with the networks as being completely defined by the the network topology. However, often we have more information about the nodes in addition to their linkages. This is especially true for social networks. For example, we see many node-wise attributes like race, sex, age, nationality and so on. More importantly, these kind of attributes sometimes have effects on the network connectivity as well, at least at the level of pairs of nodes. A popular adage captures the underlying sense nicely: "*birds of a feather flock together*".

In social networks, a known phenomenon is that nodes which are *similar* in some dimensions (i.e. share the same or similar attributes) tend to get connected to each other. This phenomenon is known as *homophily* and such mixing is often called *assortative* mixing. Examples of assortative networks are abound. In citation networks, researchers are cite other researchers in the same field. Websites tend to point to websites in the same language or political views, etc. Linguistic proximity between individuals often contributes to making friends. The assortative nature of social connections has proved to be tremendously important in designing interventions in socio-technical networks. Recommendation systems like dating apps or apps for consumable products leverage this feature by inferring matches based on correlated traits.

Mathematically, the idea of assortativity in networks is captured by the correlation of some pre-specified properties of the nodes (Newman, 2003). Given that it is basically a measure of correlation, one can also generalize the idea to conceptualize negative assortativity. *Dissasortative* networks are the ones in which neighboring nodes tend to be dissimilar. Newman (2003) made a very interesting observation, which seems to hold across real-world networks, but does not yet have a general non-contextual explanation as to exactly why it holds true. The observation is that social networks (coauthorship network, email network, company directors' network etc.) are typically assortative whereas technological (internet, power grid, electronic circuits) and biological (protein interaction networks, food web, neural networks) networks are typically disassortative. In other words, star nodes tend to connect to star nodes in the social networks whereas star nodes tend to connect to non-star nodes in technological and biological networks. Knowing the underlying mechanism of this finding may open up newer possibilities in the complex systems literature.

A simple way to characterize assortativity is to analyze the node-level similarities based on degree. For example, in a network that exhibits *degree assortativity* or *degree correlation*, high-degree nodes tend to be connected to other high-degree nodes and low-degree nodes tend to have other low-degree nodes as neighbors.

Measuring assortativity

There are two ways to measure the degree assortativity of a network, both based on measuring the correlation between degrees of neighbor nodes. The first method is based on application of Pearson correlation coefficient of the degree sequence. Given that it is a correlation coefficient, its value lies between -1 to $+1$ spanning from absolutely disassortative to absolutely assortative. However, this way of capturing assortativity is possible when the trait or characteristic based of which one is trying to find similarity, can be quantified. Degree for example, is clearly quantifiable. However, sometimes the trait may not be quantifiable. For example, inter-caste marriage (or even other types of social relations) is highly prevalent in India. We cannot use the idea of degree correlations here, since caste is a qualitative idea and not quantitative *per se*. In such cases, one can use the concept of modularity.

Technically, modularity measures the degree to which nodes sharing similar traits (even if non-quantifiable) are connected. A standard approach to measure modularity is as follows. The idea is to first quantify the differences between the observed and expected fractions of edges for each pair of types, and then to sum them up. Formally, one first labels all the nodes in the network according to some pre-assigned list of labels. Given such a labeling, the modularity for an undirected network is given by

$$Q = \frac{1}{2m} \sum_{ij} (\mathcal{A}_{ij} - E_{ij}) \delta_{c_i, c_j}, \tag{6.19}$$

where $\mathcal{A}$ is adjacency matrix, $E_{ij} = \frac{k_i k_j}{2m}$ where k_i and k_j are the degrees of nodes i and j, m is the number of edges, c_i is the label of node i, and $\delta(c_i, c_j)$ is the

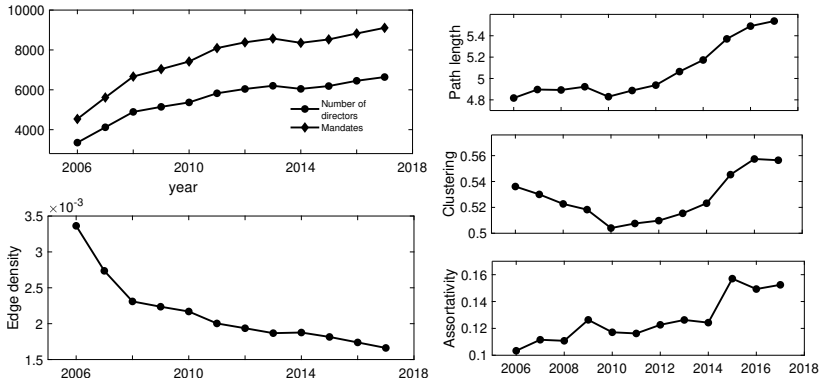

Figure 6.9 Corporate network: Evolution of the directors' network in India from 2006 to 2017. the statistics have been computed for the giant component for every year.

Kronecker delta function which equals 1 when its arguments are the same and 0 otherwise. The higher the Q, the more prominent are the modules. Note that the delta function serves as a filter over the edges, selecting only those edge-pairs $\{i,\ j\}$ whose labels denoting the traits are the same. Thus, the summation is taken only over the edges whose nodes share the same trait. The inner term is the difference between the observed fraction of all edges between i and j, which is $\mathcal{A}_{ij}/2m$, and the expected value under a null of model of a random graph with an identical degree distribution.

It is important to point it out here that modularity is a common measure of community structure, an idea that we will introduce below. The above presentation of modularity is purely based on the concept of sharing traits at the node-level. One difference between assortativity and modularity is that the earlier one is a global macro-level measure that gives a summary statistics about the structure of the network, whereas the latter one also gives a *meso*-level architecture of a network. We elaborate on the mesoscopic architecture of networks in the discussion on community structures.

In figure 6.9, we analyze the evolution of the directors' network. We see that as the network increases in size (both in the number of directors and their board mandates), the density of connections fall monotonically and the path length increases. Clustering coefficient remain more or less similar (the fluctuation is magnified due to scaling in the y-axis) whereas assortativity increases.

6.3.8 Structural balance

Structural balance is a concept at the level of a triad. Consider a four countries: A, B, C and D. Suppose A has a war with country D. Country B does not endorse that and wants to make country A lose its diplomatic ties with other countries. Country C on the other hand has an independent stance and wants to evaluate

the situation. So how would country C carry out the cost-benefit analysis? Here obviously we are ignoring all complications arising out of politics and international relations. Consider only the triad of A, B and C. In a very simple-minded way, one can see that if country C aligns with either A or B, either way it forms a stable dyadic coalition and both countries of this dyad will have a negative relationship with the other country. On the other hand if country C wants to maintain good relationships with both A and B while A and B are mutual enemies, then the triadic relationship would have a tendency to be unstable. This is essentially the concept of structural balance – the triad in the first case is balanced whereas the triad in the second case is unbalanced.

If we denote positive relationships with + sign and negative relationships with - sign, then all edges being + (i.e. + + +) or one + with two - (+ - -) would be stable configurations. On the other hand, three - (i.e. - - -) or two + and one - (i.e. + + -) would be unstable configurations.

There is a very famous result in this literature that goes by the name of Cartwright-Harary theorem (Harary, 1953). The statement of the result is as follows – Consider a network where all edges have been assigned either a positive sign or a negative sign. If the network is balanced, then there are two possibilities. One, all edges have a positive sign. Two, the network can be partitioned into two subnetworks such that within each subnetwork, all nodes edges have positive signs, but across subnetworks, all edges have negative signs.

This is a very non-trivial result. Interested readers can consult Easley et al. (2012) for a more elaborate discussion on the concept and the Harary-Cartwright theorem. There are many applications of this concept of structural balance ranging from societal connections (Antal et al., 2006) to financial comovement networks (Kuyyamudi et al., 2019).

6.3.9 External factors and their influence on linkages

Many factors external to a given network may influence its properties. Here we describe one case based on social connectedness network to elucidate such possibilities.

In figure 6.10, we plot the global connectedness network across countries based on Facebook connections. The connectedness index has been produced by Facebook researchers and the data is available from their *Data for Good* website. Bailey et al. (2018) constructed this index between the i-th country and the j-th country, based on the following definition –

$$\text{index}_{ij} \quad = \quad \frac{\text{linkages}_{ij}}{\text{users}_i \times \text{users}_j}. \tag{6.20}$$

We have plotted a thresholded version the network across 184 countries. We have also ignored self-loops. Without the threshold there would be 184^2 number of edges and the network would be incomprehensible. Equation 6.20 captures the idea of gravity equation in the trade theory which is in turn based on the Newtonian

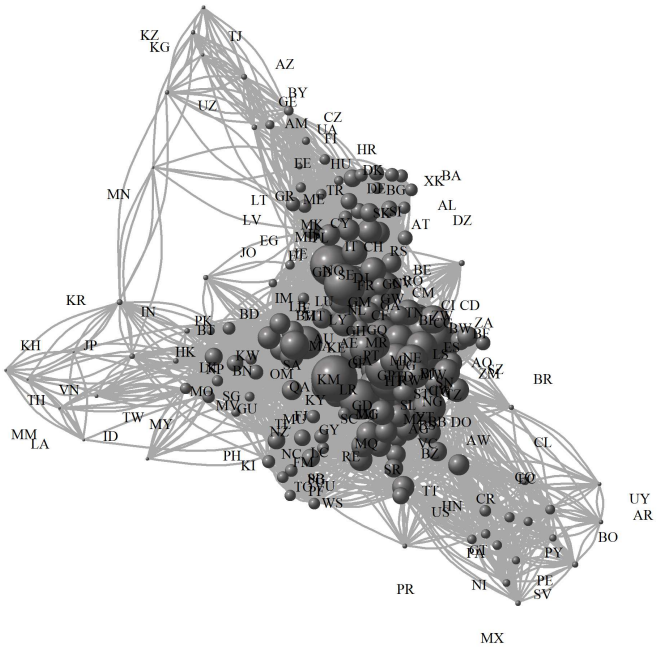

Figure 6.10 Social connectedness network across countries. Node-size is proportional to the degree of connectivity across nodes. We have applied a threshold on the edge weights for the purpose of visualization. See text for explanation.

equation of gravity. In its basic form, the gravity equation states that gravitational force between two objects is proportional to the product of masses of the objects and inversely proportional to the square of the distance between them. Equation 6.20 applies a similar idea to find the fraction of total linkages between two regions with respect to the product of the total user bases in those regions. If distance matters, then this index would vary with respect to it.

One of the most interesting findings of this network is that the effect of geographic distance on the social network is still quite prominent. When the age of social media networks began, it was thought that the world was becoming flatter and people would become connected irrespective of distance lessening the effects of geographic boundaries. In practice, what has happened is that the barrier due to geographic distance has reduced significantly, but the network as a whole has shrunk. So the

relative effect of geographic distance is still there, while the nominal effect has reduced in scope and magnitude substantially.

In a very large scale analysis based on the Facebook data, Bailey et al. (2018) analyzed the county-to-county social connectedness within the USA. They show that differences in county-wise income, population share of whites, population share of high school graduates, religious leanings and political leanings negatively correlate with social connectedness (table 2, model 5). This finding empirically captures the effect of social homophily discussed in section 6.3.7. This variable is also related to trade intensities, innovation intensities and migration across counties.

This dataset represents a very rich description of the social network globally. For further work on the same and similar dataset, see the series of papers exploring various user-level demographic variables and their interactions and effects of the social network structure (Bailey et al. (2020), Bailey et al. (2021) and Kuchler et al. (2021)).

This kind of analysis also opens up a new set of problems. The networks literature in economics and social sciences in general, confronts a problem that the realization of links are not necessarily independent of each other. The underlying link formation process (for example, whether to connect to someone or not) is not typically observable to the modeler leading to the problem that the links as observed may not be taken to be exogenous. A related problem is that in social networks, people may change their behavior if they knew the network structure of social connectivity. In other words, links can be a result of strategic choices. There is a large literature on each of these streams of work. Jackson (2010) and Goyal (2012) are two standard references for strategic games on networks. See De Paula (2020) and Chandrasekhar (2016) for reviews on the econometrics of networks. Breza et al. (2019) reviews how information flow and risk-sharing influences formation of networks in the context of economic development.

6.3.10 Specific subnetworks as building blocks

Motifs in networks

Many real world network exhibit small-scale recurring patterns that appear more than expected by chance. These patterns in connectivity are often bigger than triangles and reciprocal connections. These structures are called *motifs*. Existence of motifs indicate the presence of a localized topological structure across a large scale network. Intuitively, motifs are like bricks in a building, except that bricks are typically arranged in a much more coherent fashion.

Holland and Leinhardt are credited for the first systematic analysis of network motifs (Holland and Leinhardt, 1976, 1977). They introduced the idea of statistical likelihood of subgraphs occurring in a given network (Holland and Leinhardt, 1974, 1975). The underlying idea is that if certain types of subgraphs are more prevalent than those that can be found in random networks, then they are good candidates in a statistical sense, for being described as motifs. For a more concise description,

we can refer to Milo et al. (2002) who formalized this notion by defining network motifs as *"patterns of interconnections occurring in complex networks at numbers that are significantly higher than those in randomized networks"*.

In figure 6.11, we show some possible motifs arising out of three and four nodes in an undirected network and a directed network. In directed networks, more motifs can be generated due to the fact that the edge from node i to node j is different from the edge from node j to node i . Enumerating all motifs in a larger network (say with five nodes) will be more space consuming as the possible number of combinations will increase more than linearly. So we skip exhibiting them. Interested readers are encouraged to try to enumerate the motifs for networks with larger number of nodes on their own.

Given the concept of motifs, the next question would be how to identify them in real networks? A standard algorithm is as follows.

(i). First, enumerate all possible subgraphs of N nodes in the network.

(ii). Perform a degree-preserving randomization of the network, i.e. the links are randomized while keeping the number of nodes, edges, and degree distribution unchanged.

(iii). Next, enumerate all subgraphs of N nodes of the randomized network. Subgraphs that occur significantly more frequently in the original network as compared to the randomized analog, are called the 'motifs' of the original network.

While the idea for motif extraction is fairly straight-forward, it is quite challenging to find them. There are two major problems that crop up. One, enumerating all sub-graphs in a given network (especially if they are large) is computationally very costly. Two, assigning statistical significance to all possible sub-networks with respect to the randomized network is also difficult. However, there are many network packages in standard programming environments that are quite efficient in terms of finding motifs.

Cliques and *k*-cores

Cliques are defined as subsets of nodes in an undirected network which are directly connected to each other within the subset. In other words, cliques are complete graphs which are subgraphs of a given network (Moon and Moser, 1965). The term 'clique' originated in social sciences almost three-quarters of a century back, to study how social connections are formed (Luce and Perry, 1949).

The concept of cliques is related to k-cores in undirected networks which generalizes the notion of degree connectivity to layers of connectivity. Formally, k-core is the maximal set of nodes with the property that all nodes in the k-core have at least k links within the same subgraph. Mathematically, we write it as

$$\Omega_k = max\{i|i \in V, \sum_{j \in \Omega_k} \mathcal{A}_{ij} \leq k\}. \tag{6.21}$$

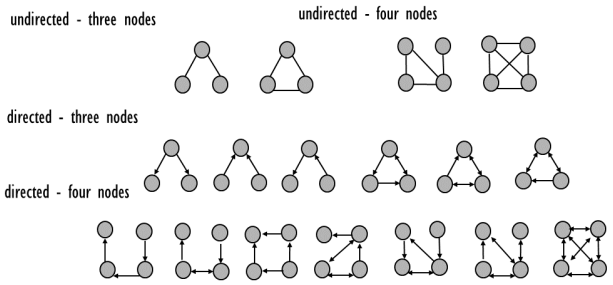

Figure 6.11 Schematic representations of key network motifs in undirected and directed networks with three and four nodes. We cannot show all possible combinations as te number of possible motives would be prohibitively large to display here. If one allows for unconnected networks, then there will be more combinations that are possible.

If we take only the layer of Ω_k which has a degree connectivity of k, we call it a k-shell.

By definition, no nodes with degree less than k can be in the k-core. The following steps lead to construction of k-cores:

(i). Begin with the original network and remove all the nodes with degree $k^{'} < k$. We call this step 'pruning' of the network.
(ii). Remove all the remaining nodes that have degree less than k after pruning.
(iii). Repeat step 2 until any further pruning leads to the degree of at least one residual node to fall below k.
(iv). The resulting network would be a k-core or a set of k-cores of the original network.

Here it is important to point out one not-so-obvious feature of a k-core. It is *not* necessary that all nodes that have a degree more than k will be in the k-core. The reason is that due to repeated pruning, it is possible that many of the connections with nodes that do not belong to a k-core will be removed, resulting in much lesser remaining degree of a given node. If the residual degree falls below k, then that node will also be removed and cannot belong to a k-core. An easy example is to consider a star-shaped network with ten nodes, one in the center and nine in the periphery. Clearly, the degree of the central node is 9. Say we want to find the 2-core of this network. A single round of pruning of nodes with degree equal to 1 leads to pruning of all peripheral nodes in the first round. The remaining node is the central node with degree zero and therefore, that will also be pruned for $k = 1$. Thus a 2-core does not exist for this network although the highest degree in the original network was much higher than k equals to 2. In figure 6.12, we show an empirical k-core decomposition of a sample of the board interlock network in 2016.

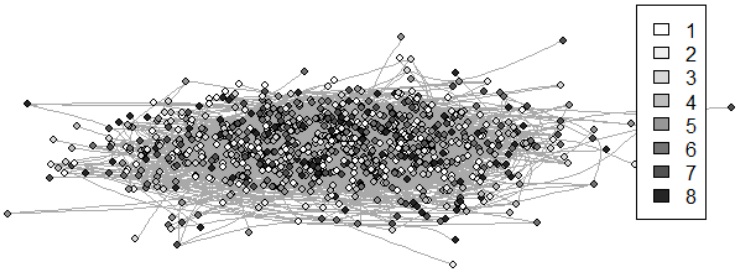

Figure 6.12 Corporate network: k-core of the Indian board interlock network in the year 2016.

6.4 Information content and filtered networks

The main reason for considering a network picture of an interconnected system is to capture the local and global topology of linkages, which presumably carries information about the nodes and/or about the network itself. However, one might ask whether it is important to consider all edges or not. The underlying point being that simple existence of an edge may not be particularly informative. First of all, what kind of information we want depends on the goal of the analysis that one would like to carry out. Secondly, even if all edges carry some information, some edges may carry more information than the others. Since the number of edges increase as square of the number of nodes in a complete network, one may want to ignore *non-informative* edges to reduce computational burden as well as to increase tractability. The question of which edge is informative depends on the definition of information and may be context-dependent. However, there are some useful context-agnostic toolkits that have been developed in the literature that allows us to extract a subgraph from a network that preserves some properties of the original network. This exercise is often called *filtering* of edges of a given network.

The obvious approach to filter edges of a weighted network is put an exogenous threshold on the weights and remove all edges that have weights below that threshold. This kind of filtering are quite common due to its simplicity. But with the advent of the literature, it has lost its appeal primarily due to two reasons. One, the threshold has to be context-dependent and most of the time, it is not possible to come up with a threshold that works for all networks. Two, given that the threshold values suffer from ad-hoc adjustments, it is not clear a priori whether the filtered network suffers from other types of loss of information (or structure) or not. For example, imposing a threshold which may otherwise seem to be accurately reflecting

the weight distribution (say, one fixes the threshold at the median of weights), it might make the network unconnected and thus the filtered network might lose the original structure. While further ad-hoc adjustments are possible, this methodology has evident theoretical and practical limitations.

With this backdrop, newer approached have been developed that are more flexible and generalizable.

6.4.1 Extreme filtering and tree structure

Trees are a special class of undirected, connected networks such that the deletion of any one link will disconnect the network into two subgraphs. The number of links in a tree with N nodes, is given by $L = N - 1$. Tree networks have derived their names from actual trees, a main property being trees have *no cycles*. As a result, for any given any pair of nodes, there is only one path connecting them. Trees are also *hierarchical*, i.e. the structure spreads out from a root node into other sub-trees. Topologically, any node in a tree can be treated as *root*. Each node in a tree is connected to a parent node (root) and to one or more children nodes (leaves). Two exceptions are: the root node (the very first node), which has no parent, and the lowest level of leaves of the tree, which have no children nodes.

Minimum spanning tree

Given an undirected and connected network $\mathcal{N} = (V, E)$, a spanning tree of the network $\mathcal{N}$ is a tree subgraph that has all the nodes V and every edge in the tree belongs to E. Generally, there can be more than one spanning trees for a given network. The minimum spanning tree (sometimes abbreviated as MST) (Cormen, 2009) is the spanning tree which has the lowest sum of weights among all the possible spanning trees.

Minimum spanning trees are useful in many different domains and have found direct applications in network design, including computer networks, transportation networks, telecommunication networks, electrical grids, and water supply networks. The theoretical importance is also immense as it also has a distant connection to the famous *traveling salesman problem*, a problem that finds the least path among a set of coordinates. While describing what is a minimum spanning tree is straightforward, how to find the tree from a given graph is less obvious. There are some well known numerical approaches that can find the minimum spanning tree quite efficiently. Below, we describe two methodologies.

Prim-Dijkstra's algorithm

Prim (1957) proposed a greedy approach to find the minimum spanning tree. While the algorithm goes by the name of Prim and Dijkstra who discovered it in 1957 and 1959 respectively, Vojtěch Jarník was probably the first to discover the algorithm in

1930. This algorithm works efficiently for dense networks. Here is a short description of the algorithm.

Algorithm 3 Prim-Dijkstra's algorithm

1: Initialize the MST with a randomly chosen node on the network.
2: Collect all the links that connect the existing tree to new nodes.
3: Find the minimum of those edges and enlarge the tree by incorporating the edge.
4: Repeat steps 2 and 3 until all nodes become part of the tree.

Kruskal's algorithm

Kruskal (1956) proposed an alternative algorithm that also became one of the standard algorithms. This algorithm works efficiently for sparse networks. The algorithm is as follows. Note that for the first edge, there does not exist a spanning tree. The

Algorithm 4 Kruskal's algorithm

1: Arrange all edges in ascending (non-descending in case of ties) order according to their weights.
2: Choose the edge with the smallest weight. If it does not form a cycle, add to the existing tree. Else, discard it.
3: Continue till the tree comprises all nodes from the original network.

first edge itself is the origin of the spanning tree. This kind of filtering is used quite frequently in case of financial comovement networks (see for example Sharma et al. (2017)).

In figure 6.13, we show an application of the minimum spanning tree algorithm on the sectoral production network. The names of the sectors are shortened for visual clarity. The original input-output network is asymmetric. We have made the network symmetric by considering the modified adjacency matrix as $\mathcal{A}_{mod} = (\mathcal{A} + \mathcal{A}')/2$ ($\mathcal{A}$ is the transpose of $\mathcal{A}$). Before applying minimum spanning tree algorithm, we carried out an element-wise inversion of the matrix $\mathcal{A}_{mod}$. This was done to ensure that the edges with the largest weights would be identified (subject to the global optimization).

6.4.2 Statistical and geometric methods

Disparity filter

Here we briefly describe *disparity filter* proposed by Serrano et al. (2009), a methodology that has steadily gained traction. The baseline idea is fundamentally linked to the thresholding approach where the threshold value is endogenous. As opposed to the rudimentary thresholding models, here the threshold is not directly applied

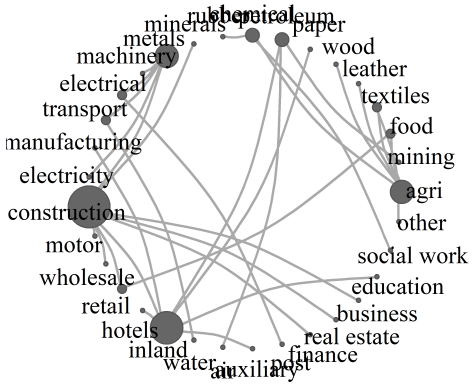

Figure 6.13 Sectoral production network: Minimum spanning tree on the inverse edge weights of the production network.

to the edge weight. Rather it is determined by statistical significance. The filtered network is shown to preserve the degree and weight distributions, and the clustering coefficient of the original network.

The algorithm works by first creating a null model of a weighted network based a random assignment of weights. Then the algorithm chooses edges and compares how likely it is a certain edge will materialize over and above random chance as compared to the null model. In particular, the null model is able to generate a distribution of weights that arise via random assignment and therefore, the algorithm can assign a p-value to the edges of the realized network. If the edge seems to be statistically insignificant based on the p-value, then the edge is removed. Otherwise, the edge is preserved.

Notationally, for a given node i in network $\mathcal{A}$, with degree k_i and sum of weighted degree $s_i = \sum_j \mathcal{A}_{ij}$, the algorithm first creates a normalized adjacency matrix with weights $\mathcal{A}_{ij}/s_i$ for all i and j. The probability that a link has weight w_i or larger is shown to be given by

$$p_{ij} = \left(1 - \frac{\mathcal{A}_{ij}}{s_i}\right)^{k_j - 1}. \tag{6.22}$$

If $p_{ij} < \alpha$, where α is a pre-specified threshold parameter that represents the desired significance level, the link is preserved, otherwise it is removed. Lower values of α lead to sparser networks, as fewer links are preserved. Higher values of α lead to lesser filtering of the network as a larger fraction of links will be deemed significant. The filtered network is often called the *backbone* of the original network. Interested readers can consult Serrano et al. (2009) for further details of the method.

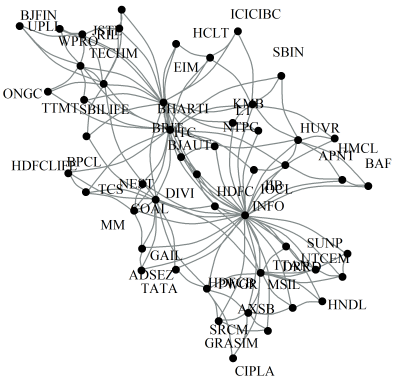

Figure 6.14 Triangulated maximally filtered graph: Applied on the network from Figure 4.4.

Filtering with planar graphs

Tumminello et al. (2005) developed a different type of filter. They utilized the concept of planar graph to filter correlation networks which contain the minimum spanning tree, but also contains more edges such that triangular loops and cliques with four nodes exist. Thus the resulting structure contains a tree structure, but itself is not a tree.

Massara et al. (2017) proposed another algorithm in the form of triangulated maximally filtered graph which approximates weighted maximal planar graph problem. In figure 6.14, we show an application of this algorithm to the return dataset that was used in figure 4.4.

6.5 Influence of nodes and edges

For real world networks, one can easily observe that all nodes are not equally important. Imagine a star-shaped network with one node in the center and a few nodes in the periphery. Intuitively, if the number of peripheral nodes increases, we may say that the node at the core becomes more *important* or *influential*. However, for analytical purposes, we have to assign precise meaning to these words and *centrality* does exactly that.

Different ideas of centrality have evolved in the literature, which convey different types of information about how important or influential nodes are in a given network. Below we describe four centrality measures that are the most well known and widely used ones.

6.5.1 Degree centrality

Probably the most obvious way to quantify centrality of a node is to consider how many edges does it have, the idea being that a larger number of edges should indicate higher level of influence of the node. In section 6.3, we had labeled this quantity as degree. Following the same nomenclature, this is often called the *degree centrality*. In case of directed graph, degree centrality can be defined both in terms of in degree and out degree. Mathematically, degree centrality $C_D(i)$ of a node i in an undirected network $\mathcal{A}$ is defined as:

$$C_D(i) = k(i) = \sum_{j=1}^{N} \mathcal{A}_{ij} = \sum_{j=1}^{N} \mathcal{A}_{ji}. \tag{6.23}$$

One can also normalize the degree centrality by the number of existing nodes in the network. This is useful for comparing degree centrality across networks with different number of nodes. The normalized centrality is written as

$$C_D^*(i) = \frac{1}{N-1} C_D(i). \tag{6.24}$$

Degree centrality is possibly the simplest measure of centrality, which is simultaneously its positive as well as negative feature. Given that it is very easy to calculate, often this measure is provided as a first cut estimate of importance of nodes. For example, social media networks like Facebook (currently, Meta) or Instagram describe how many friends do the users have. The number of friends on the social media is clearly the degree centrality. Also, degree centrality can give an idea about which nodes have very dense connections. Identification of densely connected nodes are useful for designing efficient networks for transportation. On the other hand, the same information can be detrimental if used by a terrorist group to figure out which node destabilizes the connectivity system the most.

While the ease of calculation has made degree centrality a well known measure, due to the same reason it also ignores a lot of potentially useful information. For example, by definition degree centrality accounts for connectivity only for edge length equal to 1. However, for many purposes (e.g. to model spread of a disease) one may need to account for edge lengths more than one. In particular, neighbors of neighbors may also matter and so can their neighbors. In order account for longer and more complex connections, we utilize three other measures described below.

6.5.2 Closeness centrality

Closeness centrality (C_c) measures how *close* a node is to all other nodes in a given network. In 1950, Alexander Bavelas first defined the closeness centrality (Bavelas, 1950; Sabidussi, 1966) as a reciprocal of *farness*. After almost a couple of decades, a more elaborate treatment of the idea followed (Freeman, 1978).

The idea is as follows. For a given node, first find out the shortest path (the distance) between that node and all other nodes. Then closeness centrality of that

node is defined as the sum of the inverse of all distances. Sometimes it is also defined as inverse of sum of all distances. Mathematically, $C_c(i)$ of a node i is defined as:

$$C_c(i) = \sum_{j=1}^{N} \frac{1}{d_{ij}} \tag{6.25}$$

where d_{ij} is the distance between pair of nodes i and j (defined in section 6.3.3). This definition has an obvious problem that if some d_{ij} is zero, then the measure will not be well defined. To tackle such case, an alternative definition can be considered which goes as follows:

$$C_c(i) = \frac{1}{\sum_{j=1}^{N} d_{ij}} \tag{6.26}$$

Sometimes the measure is normalized to the following:

$$C_c^*(i) = (N-1)C_c(i) = \frac{N-1}{\sum_j d_{ij}}. \tag{6.27}$$

High closeness centrality is useful for percolation – e.g. for information percolation in a short span of time.

6.5.3 Betweenness centrality

Betweenness centrality (C_B) was originally proposed to quantify the communication dynamics in a social network (Freeman, 1977). The idea is to find out which nodes act as a *bridge* between the nodes in a network. This measure captures the propensity of percolation of an information from any node in the network to any other node in the network through a given node. If the given node falls frequently on the path of percolation, we assign high value of betweenness centrality to it.

Quantitatively, the measure calculates the shortest paths among all pair of nodes and then counts how many times a given node falls on these paths. In other words, it calculates the number of times a particular node falls *in-between* the shortest paths connecting pairs of nodes. The betweenness of a node i in a graph $\mathcal{N} := (V, E)$ with N nodes, is computed as follows. First, for each pair of nodes $(s,\ t)$, compute the shortest path(s) between them. Next, for each pair of nodes $(s,\ t)$, compute the fraction of shortest path(s) that go through the node i. Finally, sum this fraction over all pairs of nodes $(s,\ t)$. Mathematically, $C_B(i)$ of a node i is defined as:

$$C_B(i) = \sum_{s \neq i \neq t \in N} \frac{\phi_{st}(i)}{\phi_{st}} \tag{6.28}$$

where ϕ_{st} is total number of shortest paths from node s to node t and $\phi_{st}(i)$ is the number of all paths that go through i. This measure can be normalized as

$$\begin{aligned} C_B^*(i) &= \frac{2}{(N-1)(N-2)} C_B(i) \\ &= \frac{2}{(N-1)(N-2)} \sum_{s \neq i \neq t \in N} \frac{\phi_{st}(i)}{\phi_{st}}. \end{aligned} \tag{6.29}$$

This measure is useful to find individuals who influence the flow of information in a network.

6.5.4 Eigenvector centrality

Finally, we introduce the idea of eigenvector centrality which gave rise to many other notions of centrality in the literature. This idea has become the cornerstone of the network theory due to its elegant algebraic structure as well as connections to diffusion process on networks. We will skip the discussion on the connection to diffusion as it goes beyond the scope of the current chapter. It suffices here to note that this measure of centrality is a key ingredient of the famous PageRank algorithm that Google uses in its search engine. However, this measure goes far back in terms of its applications. In 1972, Bonacich introduced a version of the measure to quantify prestige (Bonacich, 1972, 1987, 1991).

The idea behind this measure of centrality is that if a given node is connected to other nodes with higher centrality, then the given node will also have higher centrality. Note the roundabout nature of the definition. We did not describe exactly what is centrality of a given node here, but none the less described it to be related to centralities of its neighbor. Note that a given node is a neighbor to its own neighbors. Therefore, the centrality of a given node depends on its neighbors' centralities, which in turn is also dependent on the given node's centrality. Thus, the definition creates a self-referential measure of centrality which can be solved algebraically. The name comes from the fact that the solution to this self-referential system of linear equations is given by eigenvectors of the adjacency matrix.

Mathematically, eigenvector centrality C_e of node i in a network $\mathcal{A}$ is proportional to the sum of centrality of node i's neighbors:

$$C_{ei} = \frac{1}{\lambda} \sum_j \mathcal{A}_{ij} C_{ej} \tag{6.30}$$

where λ is a proportionality factor. This can be rewritten as:

$$\mathcal{A} C_e = \lambda C_e. \tag{6.31}$$

In the above equation, λ can be recognized to be an eigenvalue of $\mathcal{A}$ and C_e is the corresponding eigenvector. One question arises here regarding which eigenvector to consider as the candidate solution. The convention is to take the dominant eigenvector to represent the centrality measure. This also owes its mathematical support from the *Perron–Frobenius theorem* which states that an adjacency matrix with positive real values, always has a unique largest eigenvalue and all elements of the corresponding eigenvector are also positive. This largest eigenvalue is often called the *Perron–Frobenius* eigenvalue. This theorem has been extended to the case of nonnegative irreducible matrices as well.

The way we have treated eigenvector centrality above is static in nature. One can also provide a dynamic interpretation of the same idea. Specifically, one can utilize the convergence of a Markov chain to capture the idea of eigenvector centrality. We

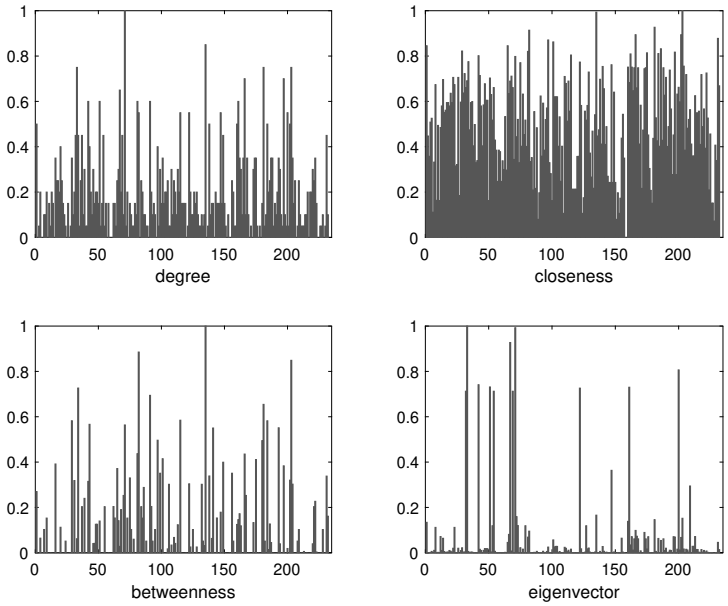

Figure 6.15 Centrality measures of the company directors in the connected component of the corporate network shown in Figure 6.6. Bar plots of (top-left panel) degree, (top-right panel) closeness centrality, (bottom-left panel) betweenness centrality, and (bottom-right panel) eigenvector centrality. As can be seen in the figure, these measures may not have particularly high correlations with each other. For each of the panels, we have normalized the maximum centrality to one to scale and visualize the y-axis in an identical fashion.

will skip the discussion here. Interested readers can consult Thurner et al. (2018) which provides the derivation of the centrality measure as a limit of a dynamic process.

The above four measures are the most elementary forms of centrality. In figure 6.15, we show all four measures of centrality constructed from a subset of the Indian corporate network shown in figure 6.6.

The literature on centrality is quite large and diverse. While it is not possible to list all possible measures of centrality here, we will mention two very important measures here. Both of them depend on the spectral structure of the adjacency matrices (like eigenvector centrality). The first one is called PageRank. The claim to fame for this came from Google's application of this measure in webpage searches and ranking. This measure builds on the idea that people searching webpage may hop from one page to another with some probability and more incoming edges increase the probability of a given webpage being visited. It can be shown that this process can be written as a diffusion process with a damping parameter and the resulting centrality vector is a transformation of the eigenvector centrality. The other one that has become very important in social networks is diffusion centrality

(Banerjee et al., 2013). This centrality measure builds on the question of which node can act as an injection point for diffusion of a information (e.g. a new product/innovation)? The more the injected information is repeated to a given node with some probability, the more likely it is for the node to be influenced by the news. Banerjee et al. (2013) show that one can construct a measure of centrality based on this idea of diffusion that is mathematically well defined under some conditions and it interpolates between degree centrality and eigenvector centrality (asymptotically). Interested reader can refer to Bloch et al. (2019) for a review of centrality measures (see also Das et al. (2018)).

One important point to note here. A natural question might arise as to if all the centrality measures effectively convey the same information or not. The answer is – no. These measures do not generally exhibit high correlations although in some case they may (see Valente et al. (2008); see also Li et al. (2015)). The other question is how stable are these measures across samples drawn from the same population. Costenbader and Valente (2003) carried out some important work in this domain. The general consensus has been that stability of the measure is often network- and more broadly, context-dependent.

6.5.5 Robust networks

By borrowing our intuitions about centrality measures, we can extrapolate the idea to conceptualize 'robustness' of a network. Whereas centrality captures the idea of how important is one node in a network, robustness captures the idea of how important is an edge in the network. The idea of robust networks is fundamentally related to the functionalities. Consider a flow network (say, a transportation network) which allows flow of entities across the nodes through its edges. Now imagine one edge of the network breaking down. In order to ensure the same flow of entities, the load that was flowing through the edge that got broken, has to be re-routed through other existing edges. But such re-routing increases loads on the residual edges and might lead to collapse of more edges, which in turn might lead to collapse of more edges due to re-routing and so on. Thus, the network might show a *cascading failure*. Robustness is defined as a property of the networks, where the extent of such cascading collapse is lesser. A more general idea of robustness would be to consider removal of nodes as well as the corresponding edges.

One can define robustness in a contextual way. One way to measure robustness in an abstract fashion is to consider by how much some global statistics for a network will change if a sequence of edges are removed from a given network. Typically, the size of the largest component is a network is taken to be a candidate measure for the global statistic. An associated measure if to consider the average distance between nodes in the components. To measure the extent of disruption, one starts from a given network and keep on removing nodes either randomly or based on hubs once at a time and compute the relative size of the giant component (i.e. the ratio of the number of nodes in the giant component to the number of nodes initially present in the network). Obviously, the network has to be connected at the

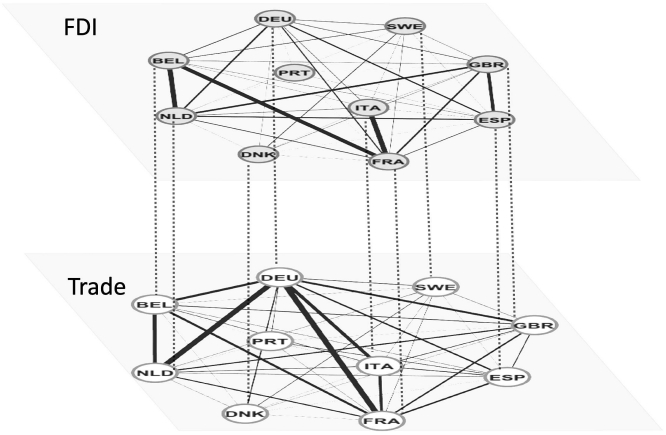

Figure 6.16 Graphical representation of multilayer network with two layers, Foreign Direct Investment (FDI) and Trade, between ten European countries in the year 2008 (FRA, ITA, DEU, SWE, GBR, ESP, NLD, PRT, BEL, and DNK). The nodes represent the countries and the links among these nodes show the FDI and Trade intensities. The width of the edges represents the strength of the relationship.

beginning. At each time step, if the removal of a subset of nodes does not break the network into disconnected components, then the proportion of nodes in the giant component would decrease moderately. If, however, the network breaks into two or more connected components, the size of the giant component will drop substantially. As the fraction of nodes removed approaches one, a few connected components will be left in the network. Hence, the proportion of nodes in the giant component will approach zero. The literature on robust network is quite large and varied. For a mathematical model interpolating between fragile and robust networks, readers can refer to Moreira et al. (2009).

6.6 Multiple layers of connectivity

So far we have dealt with networks which have only one type of edges linking the nodes. But, nodes might have multiple types of links. For example, individuals can be embedded in a collection of social networks which is combination of linkages via friendship, family ties, coworker ties etc. If we account for these multiple *layers* of connections with same type of nodes but different kind of relationships within them and across layers, then we call such networks *multilayer networks*. We show an example in figure 6.16. This shows the trade and foreign direct investment networks across a set of European countries. The source of data for this figure is Thomson Reuters Eikon Database.

Multilayer networks consist of more than one layers of networks, where nodes in each layer are both *intra-connected* and *inter-connected* (Bianconi, 2018; Demeester et al., 1999; De Domenico et al., 2013; Kivelä et al., 2014; Boccaletti et al., 2014; Porter, 2018). A special case of multilayer network is *multiplex network* which consist of more than one layers of networks with the same set of nodes (Battiston et al., 2014; Mucha et al., 2010; Gomez et al., 2013; Menichetti et al., 2014; Nicosia et al., 2013; Mishra et al., 2021). In general, each layer can encapsulate information on very different types linkages and the nodes may also differ across layers. In such cases, we call the object *network of networks*.

Evolution of links

Many real world networks are dynamic in nature. Often the same set of nodes form links to other nodes that change over time in intensity and directions. For example, the same set of Facebook (currently, Meta) friends may communicate to each other with different intensities across time. The same set of research collaborators may co-author research papers with different sub-groups of the same group of collaborators across time. For financial markets, assets may show time-varying pairwise correlations across time. Intuitively, such systems can be represented as a multiplex networks, where the same set of nodes keep on evolving in terms of the nature of their connections. We describe such multiplex networks that evolve following the arrow of time, as *temporal* networks, where each snapshot can be interpreted as one layer of the corresponding multiplex network (Holme and Saramäki, 2012, 2019; Li et al., 2017).

Fig 6.17 shows the temporal evolution of the thresholded correlation network from figure 4.4. Here we show two snapshots of the same set of companies over two consecutive days. The network changes as the nature of interaction changes. One way to capture the change in the network is to see the changes in some centrality measures of the nodes. In the panels below, we have plotted the corresponding eigenvector centralities (on the un-thresholded network; thresholding may by itself change the centrality measures drastically). The correlation measure of the two days' centrality vectors is 0.73.

The above exercise is generalizable. Since each of the layer of a temporal network is a static network, all measures that we have defined earlier for static networks, can be applied to temporal networks, with a time index. For example, the centrality measure might show time variation as the dynamic links appear and disappear as opposed links changing their intensities (as in figure 6.17).

6.7 Communities and how to detect them

Communities are meso-level structures of networks. Intuitively, the concept of communities in a network is related to different groups of nodes such that within group

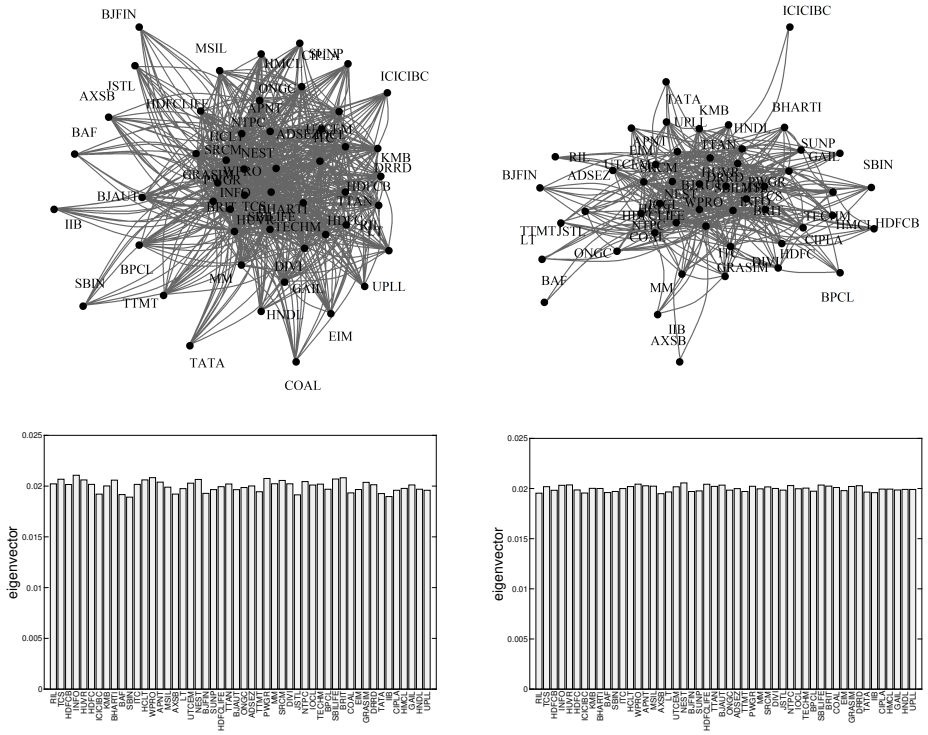

Figure 6.17 Temporal evolution of financial networks. The top-left panel reproduces the network from Figure 4.4. The top-right panel produces the network from the next day's data with the same set of companies. The network evolves with changes in the relative centrality of all the nodes. We have plotted the corresponding eigenvector centralities from the un-thresholded networks in the panels below.

connection density is high, but across group connection density is low (Radicchi et al., 2004). Much of the literature around communities in networks, have revolved around quantifying the thresholds beyond which we can call the density high (or low, for that matter, in the reverse direction). Before getting into the details, let us mention here that in the following we do not differentiate between communities and clusters. While sometimes a distinction is made, we will follow Fortunato and Hric (2016) in describing them in exactly the same way. Therefore, in our parlance, clusters would also have the same defining characteristics as communities.

Community have two defining properties (Barabási, 2016). One, communities are connected subgraphs. Two, communities have relatively dense connections. The nodes belonging to a community have a higher probability to link to the other nodes within that community than to nodes in other communities. Communities can be defined in following ways:

- *Strong community:* A subgraph $\mathcal{N}^{sub}$ whose nodes are more connected within the

community subgraph than with the rest of the network:

$$k_i^{in}(\mathcal{N}^{sub}) > k_i^{out}(\mathcal{N}^{sub}) \ \forall i \in \mathcal{N}^{sub}. \tag{6.32}$$

- *Weak community:* A subgraph $\mathcal{N}^{sub}$ is a weak community if the sum of all degrees within $\mathcal{N}^{sub}$ is larger than the sum of all degrees toward the rest of the network:

$$\sum_{i \in \mathcal{N}^{sub}} k_i^{in}(\mathcal{N}^{sub}) > \sum_{i \in \mathcal{N}^{sub}} k_i^{out}(\mathcal{N}^{sub}). \tag{6.33}$$

Recalling the definition of cliques from section 6.3.10, we note that each clique is a strong community, and each strong community in turn is a weak community. The converse is not necessarily true .

While the ideas of communities are intuitive, finding communities in large-scale networks is a challenging task. There are many methods for community detection. We will describe some of the traditional methods here. Some of the algorithms are described in the chapter on machine learning (chapter 5), as often they go beyond networks and are more generally applicable. For reference, one should consult Fortunato and Hric (2016) for a very detailed exposition of different methods for community detection and their backgrounds.

6.7.1 Two methods

Below we describe two important techniques for finding communities. The first one is based on spectral structure and the second one is based on edge betweenness. Both are applicable to real world homogeneous networks. Huang and Gao (2014) discusses a larger set of methods to apply on heterogeneous network as well.

Spectral partitioning

The first approach we will consider is called spectral partitioning or spectral bisection method. This method is based on the spectral structure of the graph Laplacian matrix. Since the Laplacian is a fundamental construct of a network, this method gained tractions in the early days of the literature when community detection *per se* received relatively less attention. Later on many better methods were developed. However, we will first review this method as this is one of the most basic methods that exploit the spectral structure of a graph and it also forms the core for many other algorithms that were developed later on.

Consider a network with adjacency matrix $\mathcal{A}$ such that the network is undirected and unweighted. Let us first define the graph Laplacian as

$$\mathcal{L} = D - \mathcal{A} \tag{6.34}$$

where D is a diagonal matrix with the node-specific degrees on the diagonal. Clearly the row and the column sums of the matrix $\mathcal{L}$ would be zero. A property of the Laplacian matrix is that it will have a zero eigenvalue with eigenvector given by $\mathbf{1} = (1, 1, ..., 1)$. This observation contains the main idea of finding communities

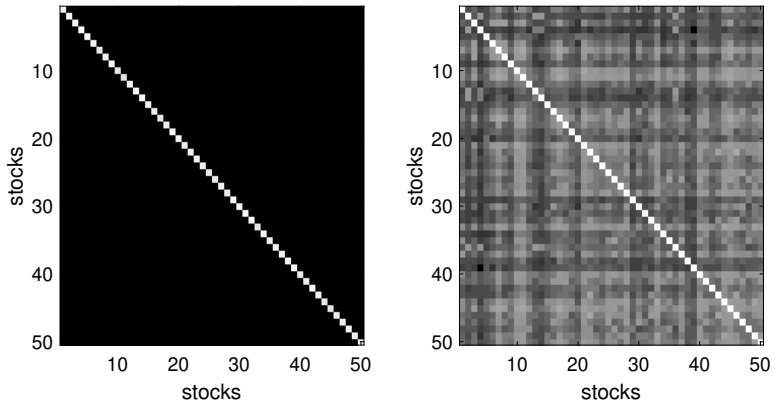

Figure 6.18 Two projections of the Laplacian of the network shown in Figure 4.4. The left panel shows the color scaling with respect to the whole matrix whereas the right panel enhances the color scaling to show heterogeneity in the off-diagonal elements.

that if a network is perfectly separated in a number of communities with no cross-community linkages, then the Laplacian would be block-diagonal and each of the block in turn would have the same property of having zero eigenvalues along with $\mathbf{1}$ as eigenvectors of commensurate size. Therefore, for a block-diagonal Laplacian, the number of eigenvalues that would be zeroes, would correspond to the number of communities. However, real networks would rarely show perfect split and in general, would contain some cross-community linkages. It is also to be noted that if the original network does in fact show perfect split across communities, then separating them would be very easy and one would not require very sophisticated techniques to begin with. In case some off-diagonal links are present connecting the communities, then typically there would be one zero eigenvalue and $s-1$ other eigenvalues corresponding to total s number of communities, that are very small in magnitude and positive. Then one can ignore the other eigenvalues and reconstruct the Laplacian by considering these s number of eigenvalues and the corresponding eigenvectors.

While it seems quite straight-forward, in practice this algorithm works well for cases where the original network has only two communities to begin with. A larger number of communities or a relatively large fraction of cross-community linkages contaminate the results and make it quite difficult to infer the true community structure. For example, this method will completely fail in cases of correlation-based networks where the off-diagonals are non-zero and there is no evident community structure a priori. In figure 6.18 we plot the Laplacian of the network shown in figure 4.4. The off diagonals cannot be classified easily in a binary fashion. Therefore, the later developments in the literature attempted to introduce newer and different tricks to bypass these problems.

Girvan–Newman algorithm

Girvan and Newman approached the problem of finding communities in a very different way (Girvan and Newman, 2002; Newman and Girvan, 2004). Their main idea is as follows. Fundamentally, the goal of community detection is to find out dense subgraphs within a given network such that the cross-community linkages are relatively fewer. They exploit the idea that just like betweenness centrality for nodes, one can create an analogous betweenness centrality for the edges. If a large fraction of shortest paths across pairs of nodes contain a given edge, then the edge has high betweenness centrality. This immediately suggests that the edges connecting communities would have high edge betweenness centrality (relative to the ones within the communities). The algorithm is quite simple. First, one can enumerate all edges in a network and compute their edge betweenness centrality. Then the edge with highest edge betweenness centrality is removed and the edge betweenness centralities of the residual edges are re-computed. If one keeps on doing it, then eventually all edges will be removed. At any intermediate stage, one can get a community structure (with varied quality). Here we provide a pseudo-code directly borrowing the steps from Girvan and Newman (2002).

Algorithm 5 Girvan–Newman algorithm

1: Compute the betweenness centrality of all existing edges in the network.
2: Remove the edge with the highest edge betweenness. In case of ties, remove one edge randomly.
3: Recompute the betweenness of all edges after the removal of this edge.
4: Keep on repeating the above two steps until all edges are removed.

This algorithm has a problem that it does not give any idea as to how many communities should be there. One can stop at any stage of the edge removal process and call the remaining network to be the designated communities. Girvan and Newman noted this problem and defined an optimal community structure based on maximum modularity to tackle this problem. Using the notation following Newman (2004b), let us define a matrix e whose generic element e_{ij} denotes the fraction of edges in the original network that connecting nodes in community i to the nodes in community j. They defined a modularity measure Q as

$$Q = \ Trace(e) - ||e^2||, \tag{6.35}$$

where the Q value represents the difference between the realized fraction of within-community edges and the expected fraction which would arise in a random graph with the same degree sequence. The maximum value of modularity across possible splits of the original network into different communities, provide a natural candidate for optimally chosen community structure. This principle of modularity maximization is more generally applicable than the baseline Girvan-Newman algorithm.

There are many other algorithms that have been proposed in the literature. Notably, hierarchical clustering has been quite prominent in applications to various

types of networks. We describe some of them in the chapter on machine learning as the applications of these algorithm go beyond networks. Other algorithms like Louvain's algorithm, stochastic block model and clique percolation, can be found in the literature with different origins and different costs and benefits of applications. Interested readers may consult Fortunato and Hric (2016) which has an in-depth review of many of these algorithms.

6.8 Network architectures

So far we have treated networks as given and defined several metrics that quantify different features of the networks. For example, for a given network, we attempted to find measures like assortativity or degree distribution and so on. Here we flip the sequence and ask, can we create a network with some given properties? For example, can we generate a network which has a given form of degree distribution? See for example figure 6.19 which shows the distribution of outflows of goods and services from the sectors we considered in figure 6.7. The outflows are measured in millions of dollars evaluated at 2010 prices. Production networks tend to exhibit power law decay (Acemoğlu et al., 2012). In this case, such a decay is not visible since we have considered a coarse-grained sectoral decomposition with only 33 sectors. Even with a small number of sectors, we can see the non-trivial dispersion of outflow across sectors. At the level of firm-to-firm networks, the power law continues to hold (Kumar et al., 2021). Finding a generative model for such distributions is a challenging task.

Over the past few decades, scientists across various disciplines including physics, computer science and economics, have made tremendous advanced in terms of generative models of networks. Here it is useful to note the key factors working behind this approach to understand the networks and how it relates to complex systems. Barabási (2016) summarized the focus of this development via a few major ideas. The following text is motivated by his descriptions of these ideas. The first point to note is that networks may occur with some given properties across many different types of systems, ranging from socio-technical to biological to physical networks. Therefore, the first insight is that one may not need different models meant for networks representing different systems. The key generative process may be identical. This lends an *interdisciplinary nature* to the models. The second feature is that while the network models can be system-agnostic, fundamentally they are driven by *empirical data*. Finally, this literature combines mathematical as well as computational approaches. Thus, taken together, these generative models can be thought of as data-driven mathematical representation of real word networks in the form of minimum-ingredient models.

In the following, we specify and describe three major models which are able to capture certain real-world features which occur in many commonly observed networks. In particular, we will describe random graph models à la Erdös and

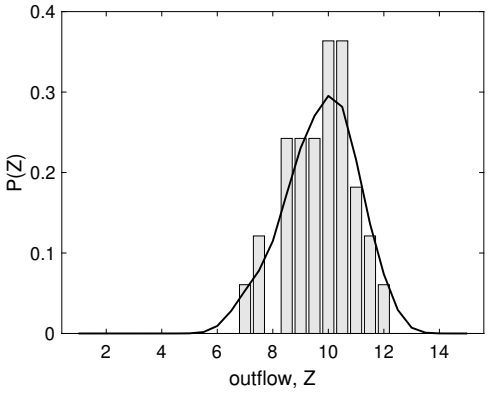

Figure 6.19 Sectoral production network: Distribution of log of outflow (out-degree) with a kernel density function fitted on the empirical histogram. We have plotted the empirical histogram for 33 sectors, which seems to be unimodal. More disaggregated datasets often exhibit a power law decay in the right tail.

Rényi, small world networks à la Watts and Strogatz, and scale-free networks à la Barabási and Albert.

6.8.1 Random graphs

This model has its origin in the work of Erdös and Rényi who introduced random graph model more than sixty years back (Erdös and Rényi, 1959). A very similar model was independently introduced by Edgar Gilbert around the same time (Gilbert, 1959).

The main mechanism behind the random graph model is quite straight forward (Newman et al., 2001). This model graph can be generated in two different ways. First, via fixed number of nodes and edges. This approach is due to Erdös and Rényi. The second approach is to fix the number of nodes and treat existence of edges to be probabilistic, due to Gilbert. Both give rise to the same class of networks.

For ease of description, we will follow the second approach. We start with N nodes and consider all $\binom{N}{2}$ pairs of nodes. For each pair of nodes, link them with some probability p. The probability parameter p lies between 0 and 1. In the following, we provide the sequential steps.

Algorithm 6 Random graphs (Erdös-Rényi-Gilbert)

1: Initiate N isolated nodes and choose a threshold parameter p where $0 < p < 1$.
2: Select a node pair $N(i, j)$ and generate a random number r such that $0 < r < 1$.
3: If $r < p$, link node i and j.
4: Repeat steps (2) and (3) for each of the $N(N-1)/2$ node pairs.

The graph on N nodes with probability p is also known as the $G(N, p)$ model as

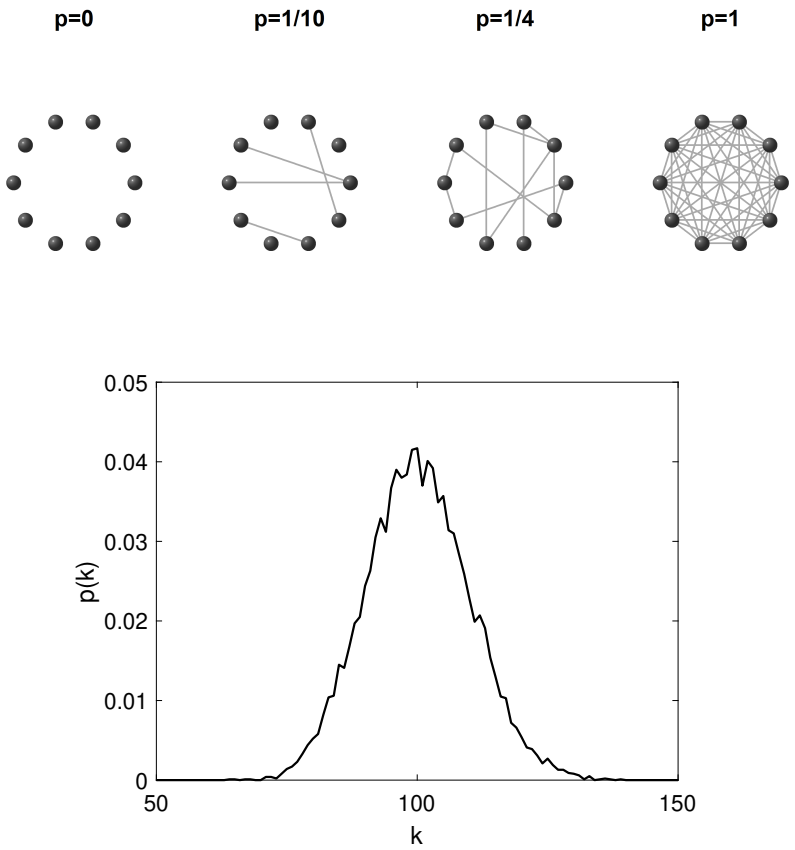

Figure 6.20 Simulation with random graphs. Top panel: Evolution of random network model of 10 nodes with growing value of p (left to right). At $p = 0$, the graph is completely unconnected. At $p = 1$, the graph is fully connected and is a complete graph. For intermediate values, fraction of edges appear probabilistically. Bottom panel: Simulated degree distribution of random network over $N = 10^4$ nodes and link probability $p = 0.01$, with 10^4 draws. As is evident, the average connectivity is numerically close to the theoretical value of Np. The curve can be smoothed further by taking average of simulated distributions.

shown in figure 6.20. Asymptotically, the difference between the Erdös and Rényi graph and the Gilbert graph, becomes negligible. The $G(N, p)$ model has well defined probabilistic characteristics. The first thing to note is that since existence of edges is probabilistic, two realizations of the $G(N, p)$ model would not have the same number of edges. The expected *number of edges* is $E = \binom{N}{2} \cdot p$. The *average degree* can be calculated as $\langle k \rangle = p(N-1) \approx pN$. The expected local *clustering coefficient* is $C = p = \frac{\langle k \rangle}{N}$. The *average path length* would be $\langle l \rangle = ln\frac{N}{\langle k \rangle}$. The

binomial degree distribution asymptotically converges to a Poisson distribution:

$$\binom{N-1}{k} \cdot p^k (1-p)^{N-1-k} \longrightarrow \frac{(pN)^k e^{-pN}}{k!}, \tag{6.36}$$

as $N \to \infty$ and $pN=$ constant.

6.8.2 Small worlds

Erdős - Rényi networks have short paths, but triangles are rare, resulting in comparatively low values of clustering coefficient. However, real networks often differ from random graphs in this dimension. In 1998, Watts and Strogatz introduced the *small-world model* which generates networks with both features – short paths and high clustering (Watts and Strogatz, 1998; Watts, 1999). This model achieved scientific stardom as it captured the famous 'six degrees of separation' phenomenon (Newman, 2000, 2003).

Let us first define the idea behind six degrees of separation. Average path length quantifies how close or far nodes are in a network (see section 6.3.3). Grid-like networks typically display high average path length as opposed to social networks which often display very low path lengths. This quantitative feature was seen in the collaboration network of Paul Erdös (of the Erdös-Rényi model), who was an extremely prolific mathematician collaborating with more than 500 scientists during his lifetime. Given his fame and this large number of collaborations, a new metric was developed informally in the name of *Erdös number*. This number is one for his direct collaborators. It is two for direct collaborators of direct collaborators of Erdös (excluding the direct collaborators themselves) and so on. Interestingly, it was noticed that many seemingly unrelated mathematicians have very low *Erdös number*. This idea of small network distance in the social networks was formalized in social networks by Stanley Milgram in the 1960s (Travers and Milgram, 1967, 1977). In his experiment, Milgram chose a group of strangers in Nebraska and Kansas (two states in the USA), and asked them to deliver a letter to target persons in Massachusetts (another state in the USA). All the participants were instructed to forward the letter to someone they knew on first name basis, who were likely to know the target. At the end of experiment, 64 letters out of 296 (roughly 21 percent) reached the target successfully and the recorded path length of reached letters was between 3 and 12 steps with the average being around 6. This led to the formation of the adage 'six degrees of separation'. This experiment has been replicated multiple times and the finding seems to be quite robust. In a study that came out in 2011, researchers examined around eight hundred million users who were actively using the Facebook platform (currently, a part of the Metaverse), with around 69 billion friendships among them, and found that the average path length was 4.74 steps which translates into 3.74 'degrees of separation' (Backstrom et al., 2012) (see also Wilson et al. (2012)).

Before getting into the description of a model for this phenomenon, we should note two points. First, the idea that social networks have a short diameter and an

even shorter average path length was qualitatively noted (at least, conceptualized) by other people as well. As Backstrom et al. (2012) notes, in the late 1920s, Frigyes Karinthy gave a concept of *Láancszemek* i.e. *Chain-Links* which essentially captures the idea that any two persons on the earth could be connected with no more than five contacts (Karinthy, 1929). Ithiel de Sola Pool and Manfred Kochen worked on a paper on the study of what they called *"Contacts and Influence"* (de Sola Pool and Kochen, 1978), which also attempted to quantify the distance between people through chains of connections quantitatively. But it would be probably fair to say that post the experiment by Milgram, and especially, given the advent of social media, the observation has been established on a sound scientific basis. While the exact number of separation varies across sample and can be a matter of debate, the fact that there is 'small world' phenomena in social networks is now a well accepted idea. Secondly, Milgram's original experiment clearly suffered from what is called *survival bias* is a statistical sense leading to potentially under-estimation of the degrees of separation. The critique is that the statistic was created out of only the successful transmissions. If one accounts for the unsuccessful transmissions (the letters that never reached), then the resulting figure would be clearly much larger and the result would be correspondingly weaker. However, as we have noted right above, the basic idea seems to have withstood the test of time successfully and has become a cornerstone finding in the social networks literature.

With the above background, let us now describe a network model for generating the 'small-world' feature. The idea behind the model is to start from a grid-like network to ensure high clustering coefficient and then to rewire some edges so that average path length is reduced (see figure 6.21). We list the steps in an algorithmic form.

Algorithm 7 Small world network model (Watts-Strogatz)

1: Construct a ring graph with N nodes, where each node is connected to its k nearest neighbors ($1 << ln\ N << k << N$) with $k/2$ neighbors on both sides.
2: For each node i, rewire every edge $\{ij\}$ such that j lie to the right of node i, to other nodes with probability p ($0 < p < 1$) such that no self-loops or duplicate edges are formed.

The expected number of rewired links in the model is $pN\frac{\langle k \rangle}{2}$. The resulting graph interpolates between two special cases: $p = 0$ corresponds to the regular network, and $p = 1$ corresponds to a random network. For intermediate values of p, the graph can have enough shortcuts to make the distances considerably small on average, but not so many as to disrupt the clustering behavior.

6.8.3 Scale-free network model

Finally, another cornerstone network structure is the class of scale-free networks which have power law degree distributions. This type of networks exhibit hubs and spokes, where hubs are highly connected central nodes. Such structures were not

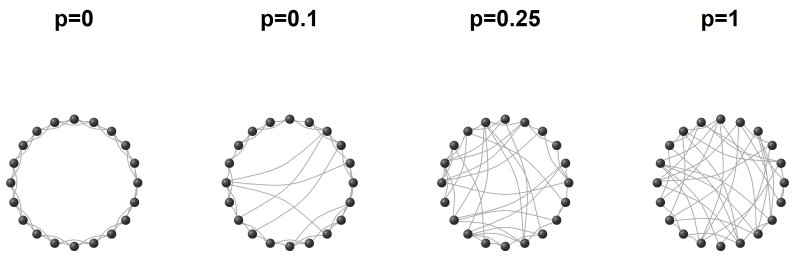

Figure 6.21 Simulation of small-world networks based on Watts-Strogatz model. A ring network with $N = 20$ nodes where each node is connected to its immediate neighbor and the neighbor of neighbor on each side. There are many triangles, resulting in high clustering coefficients for the nodes. However, paths from one node to another have to hop through many nodes, leading to high magnitude of average path length. With progressively higher probability of rewiring, the network shows small world features before eventually starting to mimic a random graph.

seen in the earlier two types of networks. The existence of hub and spoke structure is a very common occurrence in many social, biological and physical networks, where the degree distribution shows large inequality. Some nodes may have barely any connections and some nodes may have a much larger number of connections.

In our description of a generative mechanism of scale-free networks, we will follow the Barabási-Albert model (Barabási and Albert, 1999) as this is probably one of the most popular models in this literature. Their model has two crucial features, *preferential attachment* and *growth*. Historically, Yule (1925) used the preferential attachment mechanism for probably the first time to explain power-law distributions. After almost fifty years, Derek de Solla Price applied the same underlying mechanism to model the growth of citation networks constructed from scientific papers (Price, 1976). Barabási and Albert rediscovered the process independently and presented a systematic treatment of the mechanism in the context of scale-free networks.

Figure 6.22 shows examples of a network with power law degree distribution. Due to the preferential attachment mechanism, some nodes which gets more connections in the beginning, continues to probabilistically dominate nodes with lower initial degree in the subsequent steps. Eventually, it leads to emergence of large disparity in degree connectivity across nodes, which can be dubbed as a 'rich get richer' phenomenon. There are two distinguishing features of the resultant network. One, it exhibits hubs. Two, the process if dynamic and the degree distribution converges to a power law asymptotically. Mathematically, the asymptotic degree distribution

Algorithm 8 Scale-free network model (Barabási-Albert)

1: Initiate a network with (m_0) connected nodes ($m_0 \geq 2$) each node having at least one edge. For each iteration repeat steps (2) and (3).
2: **Growth:** At every point of time, add a new node to the network with $m \leq m_0$ new links attached to it. The parameter m controls the average degree of the network.
3: **Preferential attachment:** The probability π_i that a new node will be connected to node i is proportional to the degree k_i of node i

$$\pi_i = \frac{k_i}{\sum_j k_j}. \tag{6.37}$$

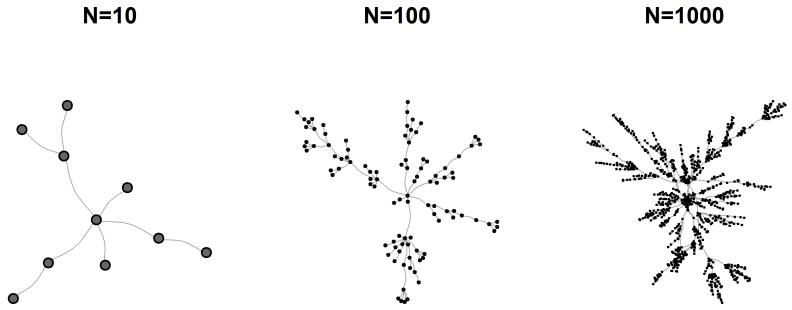

Figure 6.22 Scale-free network based on Barabási-Albert model simulated with $N = 1000$ nodes. The growth process is shown at three snapshots (N = 10, 100 and 1000).

is given by

$$P(k) \sim k^{-3}. \tag{6.38}$$

The average path length is given by $l \sim \frac{ln\ N}{ln\ ln\ N}$. If a network is characterized by the existence of a relatively large number of hubs so that the average path length scales as $l \propto\ log\ log\ N$, then we call it an *ultra-small world* network.

6.9 Shocks: Exogenous source and endogenous diffusion

Network models have been very useful in order to understand the process of diffusion. For example, how a rumor spreads or a disease spreads on a given social network is of direct and practical importance to scientists and policy-makers alike.

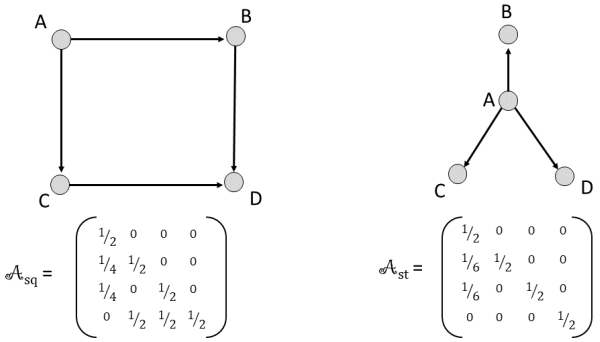

Figure 6.23 Schematic of two different networks with different implications for shock diffusion.

In the post-covid world, we are also acutely and painfully aware of the role of transportation and migration network in disease propagation.

Here we will discuss a particular type of shock diffusion in the context of production networks. This is in sync with our discussion on time series analysis in chapter 4 and the context is how an external perturbation at the node-level can spill over to the neighboring nodes and their neighbors and so on.

As a starting point we can borrow the multivariate time series framework from chapter 4, to write down shock diffusion in discrete time through the adjacency matrices in the form –

$$s(t) = \mathcal{A}s(t-1) + \varepsilon_t. \tag{6.39}$$

Consider figure 6.23 for two possible networks and their adjacency matrices. The dimension of the shock vector S at a given time point t, is 4×1 as there we have considered four nodes in figure 6.23. We assume that in the beginning, the system is in steady state indicating that $s(t = -1) = [0\ 0\ 0\ 0]'$. There are two cases we are considering captured by $\mathcal{A}_{\text{sq}}$ and $\mathcal{A}_{\text{st}}$. The first one captures a square shaped network whereas the second one captures a star-shaped network. Now, we see the effect of unit shocks on the core node A in both the networks and how they diffuse over the network via the linkages. We can simply iterate equation 6.39 by initiating the external perturbation vector as $\varepsilon_1 = [1\ 0\ 0\ 0]'$ and it is a zero vector for the rest of the time, in both cases.

We plot the resulting impulse across the nodes in figure 6.24. Clearly, the network topology plays a role in determining the extent and mode of shock diffusion. In the left panel (adjacency matrix is given by $\mathcal{A}_{\text{sq}}$), node D responds a little later than B and C as it does not directly receive the shock, and when it does, its response is larger than the rest as it receives two shocks (from B and C). In comparison,

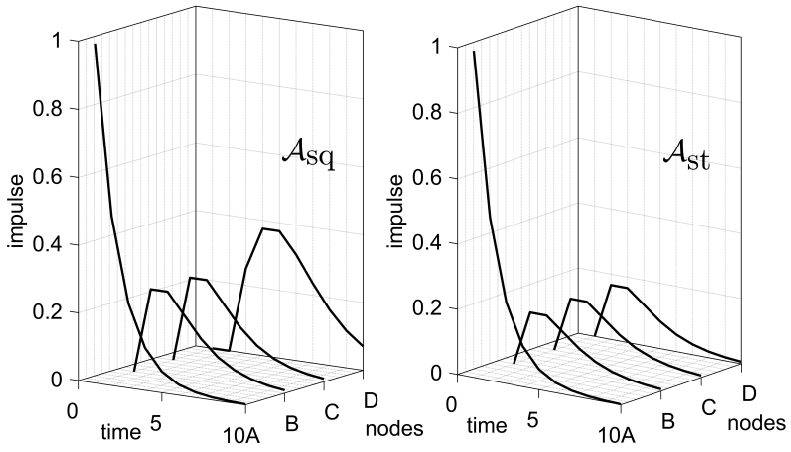

Figure 6.24 Impulse response function arising out of a shock to the core node A in Figure 6.23.

the responses are exactly identical across B, C and D in case of the star graph (adjacency matrix is given by $\mathcal{A}_{\text{st}}$). This is expected given the symmetry of these three nodes around the core node A.

6.9.1 A framework with optimization

The above kind of frame work is simplistic as model for production network. We have not considered the origin of the adjacency matrices and we also did not talk about whether the production units (firms or plants) can optimize their choices or not. Acemoğlu et al. (2012) proposed a framework which has become a cornerstone model in this literature. We will briefly review their model below.

Imagine an economy defined by a production network $\mathcal{N}$. The network has N sectors (for example, we considered 33 sectors in figure 6.7). Assume that there is a continuum of consumers with utility function over consumption of these N goods, given by

$$\mathcal{U} = \Gamma \prod_{j=1}^{N} (c_i)^{1/N} \tag{6.40}$$

where c_i denotes the consumption of the i-th sector's good. Each sector has a production function given by

$$o_j = z_j^{\alpha} l_j^{\alpha} \prod_{i=1}^{N} o_{ji}^{(1-\alpha)w_{ji}} \tag{6.41}$$

where the j-th sector produces output o_j by combining productivity shock z_j, labor employed l_j and a combination of goods purchased from all sectors $i \in \{1,\ 2,\ ...,N\}$. The out-degree sequence in this economy is given by $\{d_j\}$ where each element is

given by $d_j = \sum_{i=1}^{N} w_{ij}$. Let us define a linkage matrix W by collecting all w_{ij}. Both consumers and producers in different sectors optimize subject to their respective resource constraints.

Acemoğlu et al. (2012) showed that under a number of assumptions (competitive markets and so forth), the total output in this economy would be given by

$$\log(\text{GDP}) = v'\varepsilon \tag{6.42}$$

where $v = \frac{\alpha}{N}[I - (1-\alpha)W']^{-1}\mathbf{1}$ (I denotes an identity matrix and $\mathbf{1}$ denotes a vector of ones) and $\varepsilon = \{\varepsilon_i\}$ where $\varepsilon_j = log(z_j)$ i.e., the external shocks at the node level. Thus one can see that output of different sectors are interconnected and may respond to shocks to other sectors.

They considered the case of US sectoral input-output matrix with high resolution and showed that the out-degree distribution has a power law tail which has influence on the aggregate fluctuation of the economy's total output. In particular, higher order degree distributions also matter for shock propagation. Interested readers can consult the paper for more economic insights (see also Acemoğlu et al. (2016) for a review). While this is a static model, the idea of shock diffusion across sectors is embedded in this framework with optimization.

6.10 Taking stock and further readings

There are several excellent textbooks and monographs on network theory and applications. Interested readers can consult Menczer et al. (2020), Wasserman and Faust (1994) and Watts (1999). Newman (2010) presents a comprehensive description of the standard toolkit of network theory with an in-depth overview. Easley et al. (2012) provides a nice overview from an economic point of view. Sen and Chakrabarti (2013) presents a complementary discussion from the point of view of sociophysics. Probably, the closest reference to the current description would be Thurner et al. (2018) who describes networks in the context of complex systems.

Over the past couple of decades, the literature on networks has seen explosive growth with many new theories, applications and extensions. The study of networks has provided a useful conceptual toolkit to characterize crisis-prone systems like financial systems and has provided new ways to evaluate systemic risks (see for example Battiston et al. (2012), Battiston et al. (2016), Kuyyamudi et al. (2019) among many others). A direct descendant of this literature focuses on improving resilience and robustness of networks to external perturbations. A nascent literature in economics studies how information percolates in a given social network and is being actively used to formulate policy interventions to carry out targeted informational campaigns (Banerjee et al. (2013)). Network theory has provided a different approach to understand how scientific progress occurs (Acemoglu et al. (2016)) and how that leads to paradigm shifts eventually influencing growth and productivity of the global economy.

Some of the latest applications of network theory to the complex systems literature can be found in Sinha et al. (2010) and Chakrabarti et al. (2019). A recent stream of work on the networks literature has focused on studying networks of networks. D'Agostino and Scala (2014) provocatively described it as last frontier of complexity. The network view is not only useful, it has far-reaching implications as one of the pillars in the complex systems literature (Barabási *et al.*, 2016).