3 Hubs

hub: (*n.*) a center around which other things revolve or from which they radiate; a focus of activity, authority, commerce, transportation, etc.

If you have traveled on a plane, you have traversed an important network — the air transportation network. In Figure 0.7 we mapped the US air transportation network: the nodes represent airports and the links are direct flights between them. While most airports are rather small, few major ones (*e.g.*, Atlanta, Chicago, Denver) have daily flights to hundreds or even thousands of destinations. Similarly, in social communities there are individuals who are much more visible and influential than others; and on the Web there are some very popular sites, such as `google.com`, while most sites are unknown to most.

These examples illustrate a key feature of many networks: *heterogeneity*. Heterogeneous networks present a wide variability in the properties and roles of their elements — nodes and/or links. This reflects the diversity present in the complex systems described by networks. In air transportation networks, social networks, the Web, and many other networks, a clear source of heterogeneity is the degree of the nodes: a few nodes have many connections (Atlanta, Google, Obama) while most nodes have few.

The importance of a node or a link is estimated by computing its *centrality*. There are several ways to measure network centrality. In this chapter we introduce a few important centrality measures, for nodes in particular. As we discuss below, the degree is an important measure of centrality. High-degree nodes are called *hubs*. As it turns out, hubs are responsible for some striking properties that characterize a broad variety of networks.

3.1 Centrality Measures

3.1.1 Degree

In Chapter 1 we have learned that the degree of a node is the number of neighbors of that node. In the example of the US airport network in Figure 0.7, the degree of a node (airport) is the number of other airports reachable from it via direct flights.

In a social network, the degree of a node (individual) is the number of social links connecting the node to others. For instance, in a coauthorship network such as the one depicted in Figure 2.8, the degree is the number of collaborators. High-degree nodes in social networks are people with many connections — whether because they are sociable, sought-after, or simply eager to collaborate, these nodes seem to be important in some sense. Therefore, the degree is a very natural measure of centrality in social networks.

The *average degree* of a network indicates how connected the nodes are on average. As we shall see later (Section 3.2), the average degree may not be representative of the actual distribution of degree values. This is the case when the nodes have heterogeneous degrees, as in many real-world networks.

3.1.2 Closeness

Another way to measure the centrality of a node is by determining how "close" it is to the other nodes. This can be done by summing the distances from the node to all others. If the distances are short on average, their sum is a small number and we say that the node has high centrality. This leads to the definition of *closeness centrality*, which is simply the inverse of the sum of distances of a node from all others.

The closeness centrality of a node i is defined as

$$g_i = \frac{1}{\sum_{j \neq i} \ell_{ij}} \tag{3.1}$$

where ℓ_{ij} is the distance from i to j and the sum runs over all the nodes of the network, except i itself. An alternative formulation is obtained by multiplying g_i by the constant $N-1$, which is just the number of terms in the sum at the denominator:

$$\tilde{g}_i = (N-1)g_i = \frac{N-1}{\sum_{j \neq i} \ell_{ij}} = \frac{1}{\sum_{j \neq i} \ell_{ij}/(N-1)}. \tag{3.2}$$

This way we discount the graph size and make the measure comparable across different networks. Since what matters is not the actual value of g_i but its ranking

compared to the closeness centrality of the other nodes, the relative centrality of the nodes remains the same as by using Eq. 3.1, because the ranking is not altered if the values are multiplied by a constant. The expression $\sum_{j \neq i} \ell_{ij}/(N-1)$ is the *average distance* from the focal node i to the rest of the network. So we find that closeness can be equivalently expressed as the inverse of the average distance.

NetworkX has a function to compute the closeness centrality:

```
nx.closeness_centrality(G, node) # closeness centrality
                                 # of node
```

3.1.3 Betweenness

Many phenomena taking place on networks are based on diffusion processes (Chapter 7). Examples include the transmission of information across a social network, the traffic of goods through a port, and the spreading of epidemics in the network of physical contacts between the individuals of a population. This has suggested a third notion of centrality, called *betweenness*: A node is the more central, the more often it is involved in these processes.

Naturally, betweenness centrality has a different implementation for each distinct type of diffusion. The simplest and most popular implementation considers a simple process where signals are transmitted from each node to every other node, by following shortest paths. This approach is often used in transportation networks to provide an estimate of the traffic handled by the nodes, assuming that the number of shortest paths that traverse a node is a good approximation for the frequency of use of the node. The centrality is then estimated by counting how many times a node is crossed by those paths. The higher the count, the more traffic is controlled by the node, which is therefore more influential in the network.

Given two nodes, there may be more than one shortest path between them in the network, all having the same length. For instance, if nodes X and Y are not connected to each other but have two common neighbors S and T, there are two distinct shortest paths of length two running from X to Y: $X-S-Y$ and $X-T-Y$. Let σ_{hj} be the total number of shortest paths from h to j and $\sigma_{hj}(i)$ the number of these shortest paths that pass through node i. The betweenness of i is defined as

$$b_i = \sum_{h \neq j \neq i} \frac{\sigma_{hj}(i)}{\sigma_{hj}}. \tag{3.3}$$

In Eq. (3.3) the sum runs over all pairs of vertices h and j, distinct from i and from each other. If no shortest path between h and j crosses i ($\sigma_{hj}(i) = 0$), the contribution of the pair (h, j) to the betweenness of i is zero. If all shortest paths between h and j cross i ($\sigma_{hj}(i) = \sigma_{hj}$), the contribution is 1. If a node is

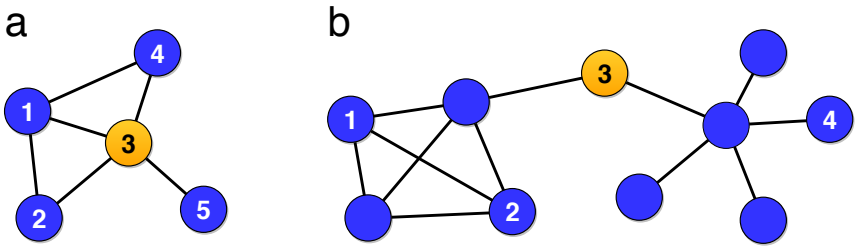

Fig. 3.1 Illustrations of node betweenness centrality. (a) The orange node has high degree ($k_3 = 4$) as well as high betweenness ($b_3 = 3.5$). (b) The orange node has a low degree ($k_3 = 2$) but it keeps the network connected, acting as the only bridge between nodes in the two subnetworks. For example, the shortest path between nodes **1** and **2** does not go through the orange node, but the path between **1** and **4** does. In fact, all the shortest paths between the four nodes in one subnetwork and the five nodes in the other subnetwork go through the orange node. Therefore its betweenness is $b_3 = 4 \times 5 = 20$.

a *leaf*, *i.e.* it has only one neighbor, it cannot be crossed by any path. Therefore its betweenness is zero. Since the potential contributions come from all pairs of nodes, the betweenness grows with the network size.

Let us work through the example in Figure 3.1(a). For node **1**, the only pair of nodes that has a shortest path going through this node is (**2**, **4**). However, there are two shortest paths of equal length between **2** and **4**: the other path goes through node **3** and not **1**. Therefore the betweenness of node **1** is 1/2. Next, consider node **3**. The shortest paths between the three node pairs (**1**, **5**), (**2**, **5**), and (**4**, **5**) go through **3**. As we observed earlier, there are two equivalent shortest paths between nodes **2** and **4**, only one of which goes through **3**, contributing 1/2 to the sum. The total gives a betweenness centrality of 3.5 for node **3**. The remaining nodes **2**, **4**, and **5** have no shortest paths going through them, therefore their betweenness is zero.

A node has high betweenness if it occupies a special position in the network, such that it is an important station for the communication patterns running through the network. For that to happen, it is not necessary to have many neighbours. Generally we observe a correlation between the degree of a node and its betweenness, so that well connected nodes have high betweenness and vice versa (Figure 3.1(a)). However, there are many exceptions. Nodes bridging different regions of a network typically have high betweenness, even if their degree is low, as illustrated in Figure 3.1(b).

The concept has a straightforward extension to links. The betweenness centrality of a link is the fraction of shortest paths among all possible node couples that pass through that link. Links with very high betweenness centrality often join to each

other cohesive regions of the network, or *communities*. Therefore betweenness can be used to locate and remove those links, allowing for the separation and consequent identification of communities (Chapter 6).

The betweenness centrality depends on the size of the network. If we wish to compare the centrality of nodes or links in different networks, the betweenness values should be normalized.

For node betweenness, the maximum number of paths that could go through a node i is the number of pairs of nodes excluding i itself. This is expressed by $\binom{N-1}{2} = \frac{(N-1)(N-2)}{2}$. The normalized betweenness of node i is therefore obtained by dividing b_i in Eq. 3.3 by this factor.

NetworkX has functions to compute the normalized betweenness centrality of nodes and links:

```
nx.betweenness_centrality(G)      # dict nodes ->
                                  # betweenness centrality
nx.edge_betweenness_centrality(G) # dict links ->
                                  # betweenness centrality
```

3.2 Centrality Distributions

Before the advent of online social media, the social networks that one could study were typically built through personal interviews and surveys, which could not involve very many people in a reasonable time frame. As a result, the networks consisted of only a few dozens of nodes. On such small networks, it makes sense to differentiate individual nodes and ask questions such as "what is the most important node of the network?" Nowadays we handle much larger graphs. For instance, the social network of Facebook friendships involves two billion individuals, including many prominent people like famous artists, sport celebrities, politicians, business people, and scientists, among others. However, no matter how popular, each of them can only be connected to a small portion of the entire network.

To better understand how centrality is distributed among the many nodes in large networks, we need to take a *statistical* approach. In this way we can focus on classes of nodes and links sharing similar features, rather than on single elements of the network. For example, we can group together all nodes having similar values of degree centrality. The statistical distribution of a centrality measure tells us how many elements — nodes or links — have a certain value of centrality, for all possible values. Figure 3.2 shows, for example, the distribution of node degree in a small network. In large networks, this is a useful tool to identify the classes of elements: by examining the distribution, we can see if there are notable values or groups of values and classify the elements accordingly. The range of the distribution also reveals the

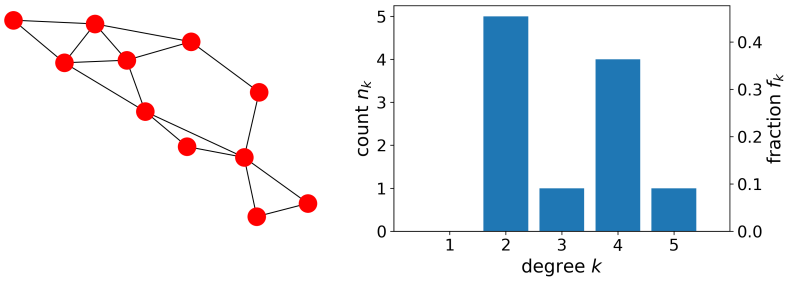

Fig. 3.2 Histogram representation of the degree distribution of a small network. First a list with the degree of each node is generated. The heights of the histogram bars are given by the counts n_k of nodes with each degree k. The relative frequency of occurrence f_k is defined as the fraction of all the nodes with degree k. The values of f_k are also shown.

heterogeneity of the network elements with respect to a specific centrality measure of interest: for example, if node degrees span many orders of magnitude, from units to millions, then the network is very heterogeneous with respect to degree. Such heterogeneity has implications both on the structure of the network and on its function, as we shall see.

Box 3.1 defines probability distributions and shows how to calculate them. To examine the probability distributions of centrality measures in real networks, let us focus on two systems: a network of Twitter users; and a network of math articles on Wikipedia (`en.wikipedia.org/wiki/Category:Mathematics`). In the Twitter network, nodes are users and a directed link from user Alice to user Bob indicates that Bob retweets (some of) the contents originally broadcast by Alice. Wikipedia nodes are pages and links are hyperlinks leading from a page to another. Both networks are directed. The Twitter network that we consider has 18,470 nodes and 48,365 links (average in-degree 2.6). The Wikipedia network has 15,220 nodes and 194,103 links (average in-degree 12.8). Such low values of average degree compared to system size indicate that both networks are sparse, *i.e.*, very few pairs of nodes are connected by links. This is a common feature of many real networks (Table 1.1).

Let us focus on the distribution of degree centrality. In Figure 3.3 we show the *cumulative degree distribution* of both networks (Box 3.1). The curves span several orders of magnitude. In such cases one says that the distributions are *broad* or have a *heavy tail*, where the tail is the right portion of the distribution, reaching to the largest values of the variable. It is customary to use the cumulative distribution when a measure has such a broad variability range. Also, heavy-tailed distributions are more effectively plotted in double *logarithmic scale*, or *log-log* scale (Box 3.2), as we have done in Figure 3.3, to be able to resolve the shape of the distribution at different orders of magnitude.

Heavy-tailed degree distributions display a large heterogeneity in the degree values: while many of the nodes have just a few neighbors, some others have many

Box 3.1 Statistical distributions

The *histogram* or *distribution* of a quantity (*e.g.*, a centrality measure) is a function that counts the number of observations (*e.g.*, nodes) having different values of the quantity. If the quantity of interest is discrete (*e.g.*, integer), for each value v we count the number n_v of observations having that value. So, the sum of n_v over all values is the total number of observations: $\sum_v n_v = N$. The result is plotted as a series of consecutive bars, one for each value, whose height is n_v.

To compare histograms of different sets of observations, it is common to divide n_v by the total number of observations N, yielding the *relative frequency* $f_v = n_v/N$. The sum of all relative frequencies is 1, regardless of the number of observations. For the node degree, the normalization is obtained by dividing by the total number of nodes (Figure 3.2). The relative frequency f_v is then the fraction of nodes with degree v.

In the limit of infinitely many observations, f_v converges to the *probability* p_v that an observation takes value v. In this limit, the histogram becomes a *probability distribution*. Any real-world network has a finite number of nodes and links, so it is impossible to reach the infinite limit, and the histogram is only an approximation of the probability distribution. However, if the network is large enough, say millions of nodes, we can treat it as a probability distribution for practical purposes.

While some centrality measures, such as the degree of a node, take integer values, other do not. For example, betweenness centrality values are not necessary integer. In these cases, instead of counting observations for specific values, we can divide the range of values into disjoint intervals, or *bins*. Then we can similarly count the number of observations falling within each bin. This binning technique can be used whenever we are interested in ranges of values, even if the values are integer. For instance, a histogram of individual wealth may count how many individuals have a yearly income within brackets, like \$0–50k, \$50k–100k, \$100k–200k, and so on.

The complementary cumulative distribution function, or simply *cumulative distribution* $P(x)$ of a variable gives the probability that an observation has a value larger than x. To compute $P(x)$, we sum the relative frequencies (or probabilities) of all the values of the variable to the right of the value x: $P(x) = \sum_{v \geq x} f_v$. The cumulative distribution is often used when the range of variability is very broad, as is the case for several centrality measures in real-world heterogeneous networks. Since high values of the variables are rare, the standard distribution has a noisy tail. The cumulative distribution effectively averages out the noise.

Box 3.2 The logarithmic scale

When plotting a curve that includes very small and very large values on one or both axes, differences between small values are indistinguishable. A solution is to plot in *logarithmic scale*: instead of using the original values as axis coordinates, we use their *logarithms*. This way a large range of values spanning many orders of magnitude can be represented effectively: small differences are amplified in the range of small values and large differences are compressed in the range of large values. We use the logarithmic scale to plot heavy-tailed distributions of network centrality measures. Since both the centrality values and the probability values span several orders of magnitude, the logarithmic scale is used on both x and y axes. We call such diagrams *log-log plots*.

neighbours, which gives them a prominent role in the network. These nodes are called *hubs*. Many natural, social, information and man-made networks have heavy-tailed degree distributions with highly-connected hubs. One way to measure the breadth of the degree distribution is to compute the *heterogeneity parameter*, which compares the variability of the degree across nodes to the average degree.

To formally define the heterogeneity parameter κ (the Greek letter "kappa") of a network's degree distribution we need to introduce the *average squared degree* $\langle k^2 \rangle$, which is the average of the squares of the degrees:

$$\langle k^2 \rangle = \frac{k_1^2 + k_2^2 + \cdots + k_{N-1}^2 + k_N^2}{N} = \frac{\sum_i k_i^2}{N}. \tag{3.4}$$

The heterogeneity parameter can be defined as the ratio between the average squared degree and the square of the average degree of the network (Eq. 1.5):

$$\kappa = \frac{\langle k^2 \rangle}{\langle k \rangle^2}. \tag{3.5}$$

For a normal or narrow distribution with a sharp peak at some value, say k_0, the distribution of the squared degrees is concentrated around k_0^2. Therefore $\langle k^2 \rangle \approx k_0^2$ and $\langle k \rangle \approx k_0$, yielding $\kappa \approx 1$. For a heavy-tailed distribution with the same average degree k_0, $\langle k^2 \rangle$ blows up because of the large degree of the hubs, so that $\kappa \gg 1$.

If the degree distribution is concentrated around a typical value, there is no heterogeneity and the parameter is typically close to one.[1] If the degree distribution is broad, instead, the heterogeneity parameter is heavily inflated by the largest degrees of the hubs, and may take large values. The more hubs there are, the larger

[1] An alternative definition in the literature compares the heterogeneity parameter with $\langle k \rangle$ rather than with one.

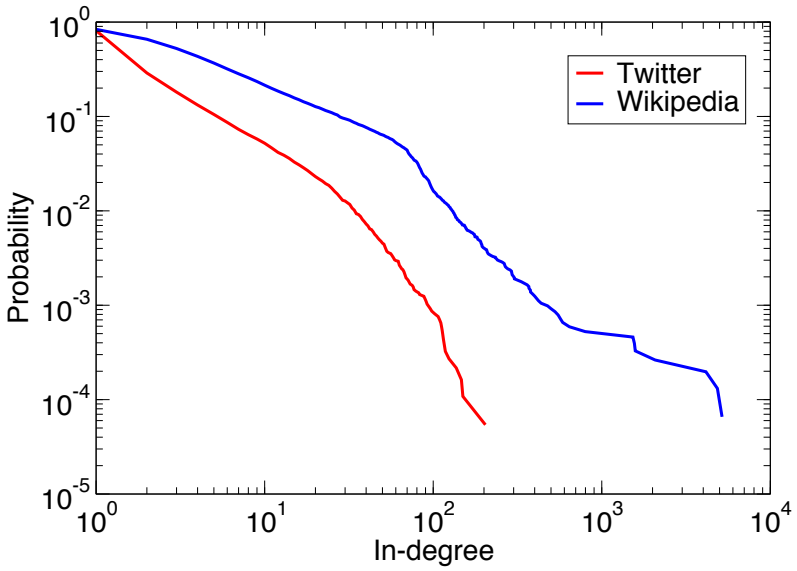

Fig. 3.3 Cumulative degree distributions of Twitter and Wikipedia networks, shown on a log-log plot. Both graphs are directed, so we show the in-degree distributions. The maximum in-degree is 204 for Twitter and 5,171 for Wikipedia. The curves are plotted using the logarithmic scale because they span several orders of magnitude.

the heterogeneity. As we shall see, heterogeneity plays a key role in the structure of a network, and in the dynamics of some processes running on it.

If a network is directed, like our Wikipedia and Twitter graphs, we have to consider two distributions, the *in-degree* and *out-degree* distributions, defined as the probability that a randomly chosen vertex has a given in- or out-degree, respectively. In this case, the definition of hub may refer to either the in-degree or the out-degree. For instance a Web page may have many other pages linking to it (large in-degree), but it may itself link to just a few pages (low out-degree), or vice versa. In several directed networks the two measures are *correlated*, so nodes with large (small) in-degree also have large (small) out-degree. We will return to the discussion of degree in directed as well as weighted networks in Chapter 4. Table 3.1 reports some basic numbers characterizing the degree distributions of various networks.[2]

One can of course analyze the distributions of other properties besides the degree. It turns out that the degree is usually correlated with other centrality measures. So,

[2] Datasets for these networks are available in the book's Github repository: `github.com/CambridgeUniversityPress/FirstCourseNetworkScience`

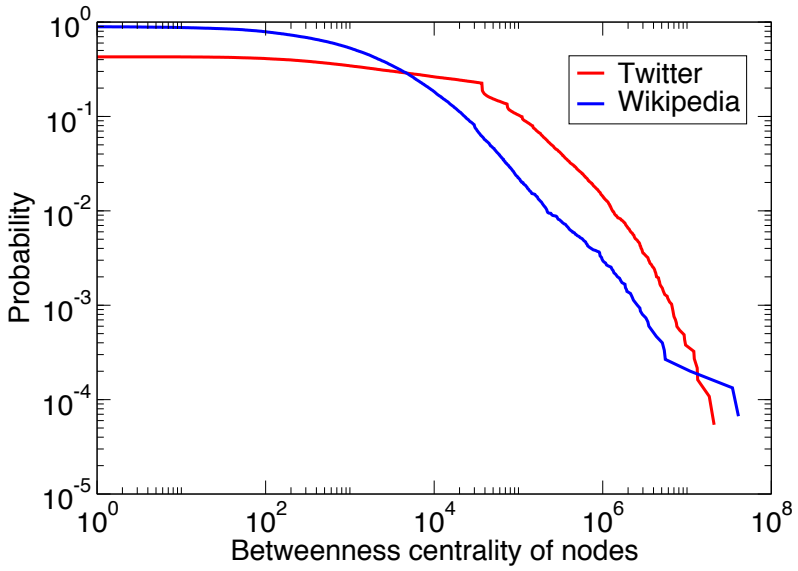

Fig. 3.4 Cumulative distribution of node betweenness centrality for Twitter and Wikipedia, shown on a log-log plot. We considered both networks as undirected. For Wikipedia we computed the betweenness only on its giant component, which includes over 98% of the nodes. The Twitter graph is connected.

hubs typically rank among the most central nodes with respect to diverse criteria. There are exceptions as well. As we have seen in Figure 3.1, a node may have a large betweenness centrality if it connects different areas of the network, whether or not it has high degree.

In Figure 3.4 we show the cumulative betweenness distributions for our networks. Just like the degree distributions, they too span multiple orders of magnitude.

Hubs, when present, are the single most important feature of a network. They are the pillars of its structure and the drivers of the processes running on it. In the next sections we present some remarkable consequences of the presence of hubs.

3.3 The Friendship Paradox

Suppose you are looking for the person who has the largest number of friends, among a group of N people for whom you only have a directory of phone numbers.

Table 3.1 Basic variables characterizing the degree distribution of various network examples: average degree, maximum degree, and heterogeneity parameter. The networks are the same as in Tables 1.1 and 2.1, their numbers of nodes and links are listed as well. For directed networks we report the maximum in-degree and the heterogeneity parameter is computed on the in-degree distribution.

Network	Nodes (N)	Links (L)	Average degree ($\langle k \rangle$)	Maximum degree (k_{max})	Heterogeneity parameter (κ)
Facebook Northwestern Univ.	10,567	488,337	92.4	2,105	1.8
IMDB movies and stars	563,443	921,160	3.3	800	5.4
IMDB co-stars	252,999	1,015,187	8.0	456	4.6
Twitter US politics	18,470	48,365	2.6	204	8.3
Enron Email	87,273	321,918	3.7	1,338	17.4
Wikipedia math	15,220	194,103	12.8	5,171	38.2
Internet routers	190,914	607,610	6.4	1,071	6.0
US air transportation	546	2,781	10.2	153	5.3
World air transportation	3,179	18,617	11.7	246	5.5
Yeast protein interactions	1,870	2,277	2.4	56	2.7
C. elegans brain	297	2,345	7.9	134	2.7
Everglades ecological food web	69	916	13.3	63	2.2

If you just call one of the numbers, chosen at random, the chance that you picked the right person is $1/N$. What if you ask them about one of their friends? It may seem that you are just selecting another individual at random, like before, and that the chance that the friend is the right person is the same. But that is not the case. To see why, consider the small social network in Figure 3.5. The most connected individual is Tom, who has four friends. If you question a random individual, there is one possibility out of seven that you pick Tom. However, if you select a random friend of a random individual, the probability that you bump into Tom turns out to be $5/21 \approx 24\%$, which is quite a bit larger than $1/7 \approx 14\%$. We conclude that it is easier to find people through their friends than by random search. But why?

Roughly speaking, if someone has many friends, she has a far greater chance to be mentioned by someone than if she had just a few. Reaching out to someone's friends actually means choosing links instead of nodes. When we go for the nodes, each of them has the same probability to be selected, regardless of their degree. When we go for the links, the larger the number of neighbors of a node, the higher the probably it will be reached. In our network of Figure 3.5 there are four possible channels leading to Tom, so it is far easier to reach him than Mary and Tara, who only have one friend.

The chances of hitting a hub increase if you move from the circle of neighbors to that of the neighbors of neighbors, and so on. This is because the number of links to follow increases at each step, so it becomes more likely that one of them is attached to a hub. This property can be used to our advantage. There are many situations in which identifying the hubs of the network could be helpful. For instance, during an

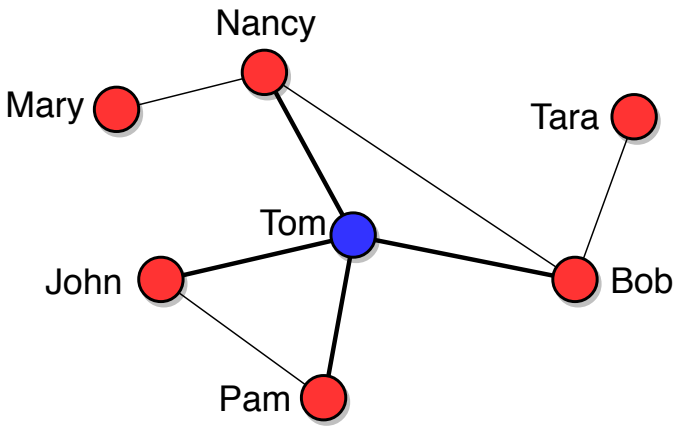

Fig. 3.5 Friendship paradox. By selecting a random link instead of a random node, Tom can be "found" much more easily than Mary, because he has four friends (John, Pam, Bob, and Nancy), whereas she has only one (Nancy). It is far more likely to bump into a hub than into a node with low degree when following connections at random. This is the underlying reason why our friends have more friends than us, on average.

epidemic outbreak, the individuals with the largest number of contacts are potential big spreaders and it would be important to isolate and/or vaccinate them to contain the disease. In such a scenario, one could select people randomly and get in touch with some of their friends, as they have a higher probability of being hubs than the pool of selected individuals. We will revisit this topic in Chapter 7.

The difference in the selection of links versus nodes has another peculiar implication. Let us choose an actor in our network, say Nancy. She has three friends: Bob, Mary and Tom. They have in total $3 + 1 + 4 = 8$ friends, which gives an average of $8/3$. If we repeat this calculation for all other nodes, we find that the average number of neighbors of the neighbors of a node is $17/6 = 2.83$. On the other hand, the average degree of the network is $(1 + 3 + 3 + 1 + 4 + 2 + 2)/7 = 16/7 = 2.29$. This is typical: the average degree of the neighbors of a node is larger than the average degree of the node. In other words, our friends have more friends than us, on average. This is known as the *Friendship Paradox*.

Our example helps us uncover the origin of the paradox. When we compute the average degree of a node, each node's degree appears only once in the sum. On the other hand, when we compute the average degree of the neighbors of a node, and we repeat the procedure for all nodes, each node will appear as many times as its degree in the partial sums. In our example, Tom's degree will be counted four times because he is in the list of friends of four people. This boosts the value of the neighbors' average degree, which ends up being larger than the average degree. The Friendship Paradox is thus due to *sampling*. The two averages are computed by

sampling the node degrees differently: uniformly for average degree, proportionally to the degree for the neighbors' average degree.

The broader the degree distribution, the stronger the effect of the friendship paradox. When all nodes have approximately the same degree, the two values are similar to each other. In networks with heavy-tailed distributions, like the ones featured in Figure 3.3 (and like typical social networks), the effect is very pronounced because of the super-connected hubs.

3.4 Ultra-Small Worlds

The hubs of a network are not only easy to find, as we have just seen; they are also in great demand. If we want to transmit a signal from one node of the network to another along the shortest route, the signal is likely to pass through one or more hubs. You have probably experienced this in your air travel: when you want to fly from airport **A** to airport **B**, and there is no direct flight between **A** and **B**, you are forced to make at least one connection at some hub airport **C**. In many cases, one connection is enough, so the trip from **A** to **B** requires only two flights: **A** $\rightarrow$ **C** and **C** $\rightarrow$ **B**.

In Chapter 2 we have seen that many real networks are *small worlds*, *i.e.*, one can go from every node to any other node with a small number of steps. In a network with hubs, we expect that the average distance between any two nodes is shorter compared to a network with the same number of nodes and links, but no hubs. In fact, networks with broad degree distributions often have the so-called *ultra-small world* property indicating that the distances between nodes are very short. In Figure 3.6 we plot the distribution of the distances between any two nodes for the reference networks we have been using: Twitter and Wikipedia. Both distributions are strongly peaked, so there is very little variability among the distances. The peak values are extremely small compared to the system sizes (five for Twitter, three for Wikipedia), indicating that both networks are ultra-small worlds. This is a trademark feature of many real networks.

3.5 Robustness

A system is *robust* if the failure of some of its components does not affect its function. For instance, an airplane keeps flying if one of its engines stops working. In general, robustness depends on which components fail and on the extent of the damage.

How to define the robustness of a network? Nodes can describe a broad variety of entities, such as people, routers, proteins, neurons, Web sites, and airports. In such a high-level representation, it is not straightforward to define the failure of a node,

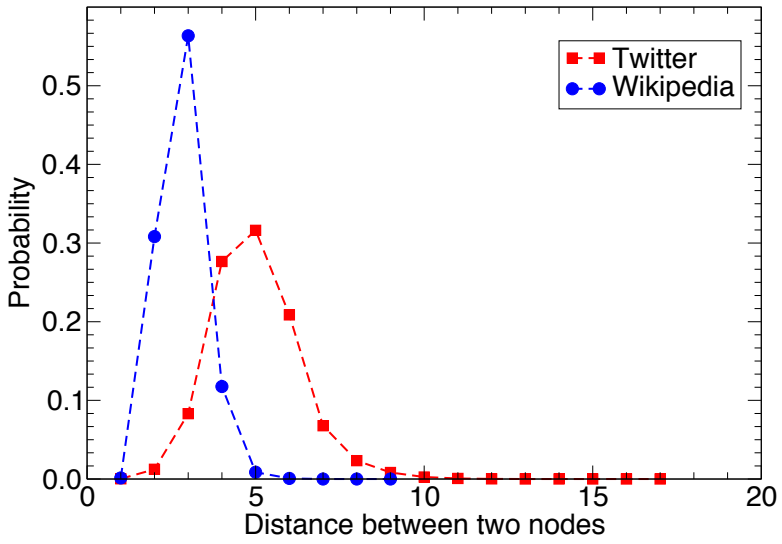

Fig. 3.6 Ultra-small worlds. The distributions of distance between nodes are peaked at very low values for both Twitter and Wikipedia. This is due to the presence of hubs, which shrink the distance between most pairs of nodes, as shortest paths run through them. Distances are computed by ignoring the direction of the links.

which depends on the specific type of network. But if we assume that a node stops working, somehow, we can ask how the structure and consequently the function of the network change without that node and all of its links.

In Chapter 2 we defined what it means for a network to be connected — all nodes are reachable from each other. We also saw that if a network is not connected, it has two or more connected components. Connectedness is an important network property that typically affects its function. If the Internet were not a connected graph, it would be impossible to send signals (*e.g.*, emails) between routers belonging to different components. Therefore, one way to define and measure the robustness of a network is to observe how the removal of a node and its links affects the connectedness of the system (Figure 3.7). If the system remains connected, we can assume that it will keep working fine, to some extent. On the other hand, a breakup of the network into disconnected pieces would signal severe damage that might compromise its function.

The standard robustness test for networks consists in checking how the connectedness is affected as more and more nodes are removed, along with all of their

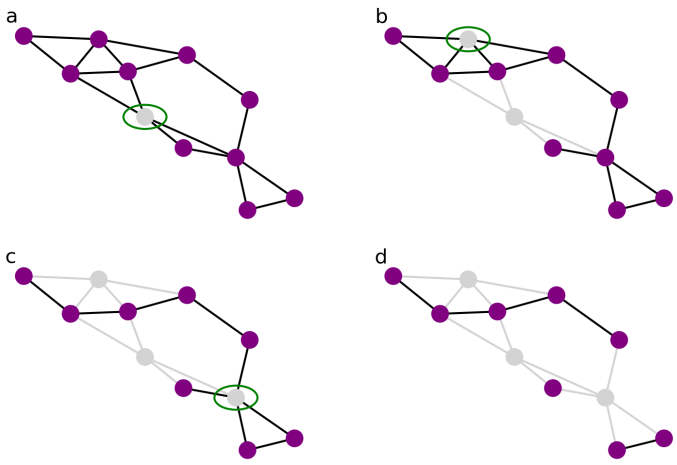

Fig. 3.7 Network robustness. Effect of a sequence of deletions of nodes and their incident links. In each diagram the deleted node is highlighted by a circle. Deleted nodes and their incident links are colored in grey. After three nodes are deleted (d), the network breaks into three components disconnected from each other.

adjacent links. To estimate the amount of disruption following node removal, scholars 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. Let us suppose that the initial network is connected. In this case the giant component coincides with the whole network, so its relative size is one. If the removal of a subset of nodes does not break it into disconnected pieces, the proportion of nodes in the giant component just decreases by the fraction of removed nodes. If on the other hand the node removal breaks the network into two or more connected components, the size of the giant component may drop substantially. As the fraction of removed nodes approaches one, the few remaining nodes are likely distributed among tiny components, so the proportion of nodes in the giant component is close to zero.

Figure 3.8 illustrates the outcomes of robustness tests on the OpenFlights World network. When nodes are removed randomly, the process simulates the *random failure* of network elements. We observe that the relative size of the giant component decreases very slowly. This is due to the presence of hub nodes, which keep the structure connected. As long as a sufficient number of hubs survives, the system remains largely connected. Since we are removing nodes at random, the probability of hub failure is low, because they are statistically rare compared to other nodes.

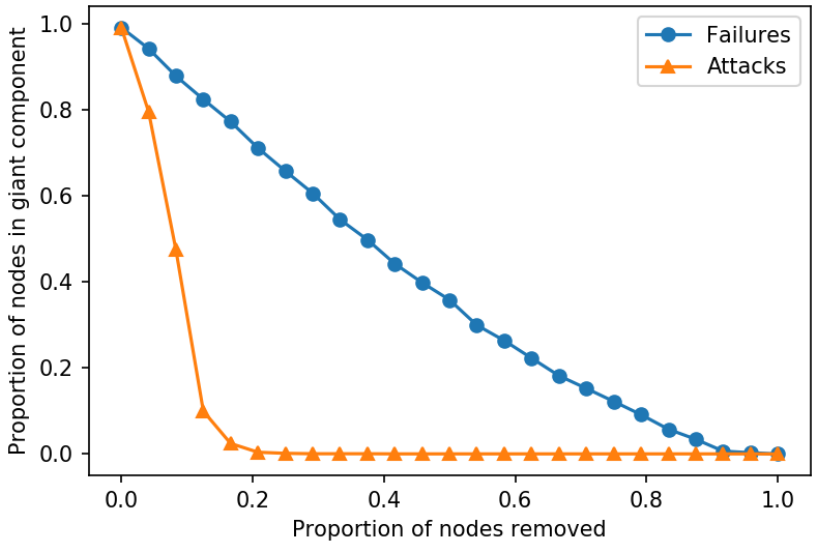

Fig. 3.8 Network robustness. Fraction of nodes in the giant component as a function of the fraction of nodes removed from the OpenFlights World network. We show what happens if nodes are removed at random (random failures), or prioritized based on degree (targeted attacks).

The plot also shows what happens when nodes are removed in decreasing order of their degree, *i.e.*, the hubs are targeted first. In this case, the system suffers a major disruption almost immediately, and is totally fragmented when about 20% of the nodes are eliminated. Targeting high-degree nodes is an example of *attack*, as one aims at maximizing damage by removing central nodes. We conclude that many real networks, which have central hubs, are pretty robust to random failures, but quite vulnerable to attacks.

3.6 Core Decomposition

We briefly mentioned the *core-periphery structure* of many networks in Section 2.1. When analyzing or visualizing a large network, it is often useful to focus on its denser portion (core).

The degree of each node can be used to separate a network into distinct portions, called *shells*, based on their position in the core-periphery structure of the network. Low-degree outer shells correspond to the periphery. As they are removed, or peeled away, what remains is a denser and denser inner subnetwork, the *core*. We start with singletons (zero-degree nodes), if there are any. Then we remove all nodes with

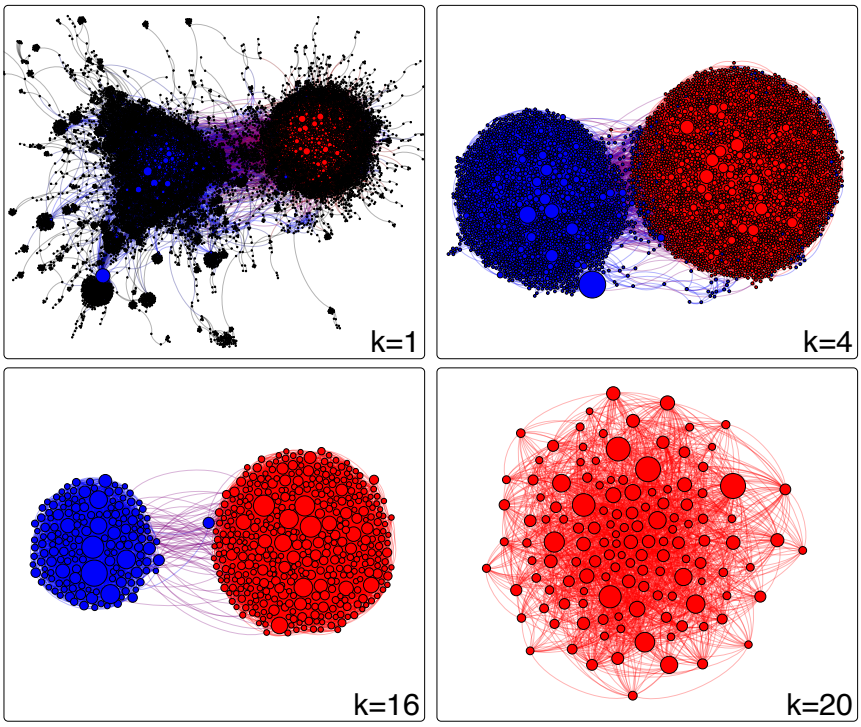

Fig. 3.9 Filtering by k-core decomposition. We start from the full Twitter political retweet network ($k = 1$). As we increase k, peripheral nodes are removed and the remaining core becomes smaller and denser. The innermost core contains only red nodes, corresponding to conservative accounts; each has at least $k = 20$ neighbors.

degree one. Once there are none left, we start removing nodes with degree two, and so on. The last group of nodes to be removed is the innermost core.

Formally, the *k-core decomposition* algorithm starts by setting $k = 0$. Then it proceeds iteratively. Each iteration corresponds to a value of k and consists of a few simple steps:

1 Recursively remove all nodes of degree k, until there are no more left.
2 The removed nodes make up the *k-shell* and the remaining nodes make up the $(k+1)$-*core* because they all have degree $k+1$ or larger.
3 If there are no nodes left in the core, terminate; else increment k for the next iteration.

Core decomposition is particularly helpful in practice for filtering out peripheral nodes when visualizing large networks. In fact, most of the figures in Chapter 0

do not depict entire networks, but only portions obtained by excluding some of the periphery. For example, the $k = 1$ and $k = 2$ shells have been removed in the political retweet network shown in Figure 0.3. This filtering process is illustrated in Figure 3.9.

NetworkX has functions for core decomposition:

```
nx.core_number(G) # return dict with core number of each node
nx.k_shell(G,k)   # subnetwork induced by nodes in k-shell
nx.k_core(G,k)    # subnetwork induced by nodes in k-core
nx.k_core(G)      # innermost (max-degree) core subnetwork
```

3.7 Summary

In this chapter we have learned about different centrality measures of network nodes and edges, and focused on the degree of nodes as an important measure that identifies hubs. A few concepts to remember:

1 The degree of a node is defined as the number of links in the graph incident on the node.

2 The betweenness of a node expresses how often it is traversed by signals propagating on the networks following shortest paths.

3 In large networks it is necessary to use statistical tools to analyze the global features of the network. The histogram provides a visual illustration of the distribution of a given attribute of nodes or links (*e.g.*, the degree). The normalized histogram is an estimate of the probability distribution of the measure of interest.

4 The distributions of centrality measures are heterogeneous for many real networks, *i.e.*, they span multiple orders of magnitude. In particular, the degree distribution often has a heavy tail. Nodes with large degree are called hubs.

5 The Friendship Paradox says that in a social network, your friends have more friends than you do, on average. This is due to the high probability of selecting hubs among a node's neighbors.

6 Hubs play a critical role in the structure and dynamics of a network. For instance, they shrink the distances between nodes and make the network robust against random failures, but vulnerable to targeted attacks.

7 We can decompose the network to reveal its core-periphery structure. This is accomplished by iteratively filtering out shells of low-degree nodes and focusing on the remaining, denser and denser cores.

3.8 Further Readings

Closeness centrality was introduced by Bavelas (1950). Freeman (1977) introduced node betweenness and Brandes (2001) developed the algorithm commonly adopted to calculate it. Link betweenness, introduced in an unpublished technical report by Anthonisse, is well described by Girvan and Newman (2002), who applied the measure to detect and remove links connecting network communities to each other, so that the latter can be separated and identified (Section 6.3.1). Statistical distributions are nicely presented in the book by Freedman et al. (2007).

An accessible introduction to networks and their hub structure is offered by Barabási (2003). Albert et al. (1999) discovered the first large network with a heavy-tailed degree distribution, namely the Web graph. Many other real-world networks were subsequently found to have the same property (Barabási, 2016).

The friendship paradox was exposed by Feld (1991). Ultra-small worlds were discovered by Cohen and coworkers (2002; 2003). The first study of network robustness appeared in a paper by Albert et al. (2000). Cohen et al. (2000, 2001) have authored classic theoretical studies on robustness.

The application of k-core decomposition to network visualization is due to Batagelj et al. (1999); Baur et al. (2004); Beiró et al. (2008).

Exercises

3.1 Go through the Chapter 3 Tutorial on the book's GitHub repository.[3]

3.2 Assume you have a graph with 100 nodes and 200 links. What is the average degree of nodes in this network?

3.3 Consider a network formed by 250 students in a dormitory. The links in this network represent roommate relationships: two nodes are connected if they are currently roommates. In this dorm, the rooms are mostly double occupancy with a few triples and quads.

1 Is this graph connected?
2 What is the mode (most frequent value) of the node degree distribution?
3 How many nodes are in the largest clique?
4 Would you expect this graph to have any hubs?

3.4 In NetworkX, how can you find a node with the largest degree centrality in a network? And how would you also get the degree of that node?

3.5 Assume you have a NetworkX graph G of employees. The node names are employee IDs, and the nodes have attributes for full name, department, position,

[3] `github.com/CambridgeUniversityPress/FirstCourseNetworkScience`

and salary. Which of the following will give you the salary for the employee with ID 5567?

a. `G.node(5567)('salary')`
b. `G[5567]['salary']`
c. `G.node[5567]['salary']`
d. `G(5567)('salary')`

3.6 You have a NetworkX graph `G` and you are about to draw it with the following command: `nx.draw(G, node_size=node_size_list)`. Which of the following is a correct way to obtain `node_size_list` so that the nodes are sized according to their degree?

a. `node_size_list = [G[n] for n in G.nodes]`
b. `node_size_list = G.degree()`
c. `node_size_list = [G.degree() for n in G.nodes]`
d. `node_size_list = [G.degree(n) for n in G.nodes]`
e. `node_size_list = [d for d in G.degree()]`

3.7 An academic collaboration network is one type of social network. In such a network, a node with degree two means that:

a. A scholar has co-authored a paper with one other scholar
b. A scholar has co-authored publications with two other scholars
c. A scholar has authored two publications
d. A publication was co-authored by two scholars

3.8 In a social network, which of the following would one expect to be true about the degrees of its nodes?

a. Most nodes connect to a single, large hub
b. A variety of degrees is to be found
c. All nodes have more or less the same degree
d. All nodes have very high degree

3.9 What property does a network have to have in order for closeness centrality to be well-defined?

3.10 Provide examples of networks such that:

1 The node with the highest degree is not the one with largest closeness.
2 The node with the highest betweenness is not the one with largest closeness.

3.11 Consider the network in Figure 3.10 in order to answer the next few questions. For each question, in case of a tie, answer with all the tied top nodes.

1 Which node has the highest degree centrality?
2 Which node has the highest betweenness centrality?
3 Which node has the highest closeness centrality?

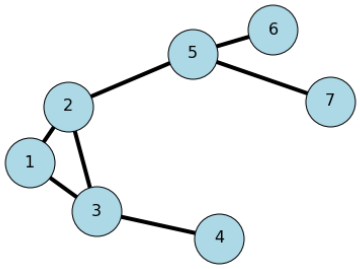

Fig. 3.10 An undirected, unweighted network.

3.12 Suppose we want to build a connected network with 10 nodes and average degree 1.8, such that the heterogeneity parameter is largest. What does the graph look like?

3.13 For each of the following variables, state whether or not you would expect to see a heavy-tailed distribution, and why:

1 The shoe size of UK adults
2 US household income
3 Node degree in the Twitter social network
4 Pairwise distance in the Wikipedia network

3.14 If people heights followed a heavy-tailed distribution, would you be surprised to see a 30-foot (9-meter) tall person in the street?

3.15 The plot in Figure 3.11 comes from a study of 200 million Web pages and 1.5 billion links between them (Broder et al., 2000). It is a log-log plot of the number of pages (y-axis) with a given number of in-links (x-axis):

1 Approximately how many pages have only one other page linking to it?
2 Approximately how many pages have ten other pages linking to it?
3 Approximately how many pages have 100 other pages linking to it?

3.16 Consider a social network where a connection represents a sexual relationship. Read the report by Liljeros et al. (2001) about a study of such a network based on a sample of 4781 Swedes. (If you do not have access to the journal through your institution, you can download a preprint of the paper at `https://arxiv.org/abs/cond-mat/0106507`.) What is the maximum degree in this network? What does it mean? If you consider the subnetworks with nodes corresponding to males and females, respectively, do they have the same degree distribution? Why or why not?

3.17 A common use of the word "hub" in everyday speech is to describe airports that serve many routes (direct flights). Load the OpenFlights US flight network into a NetworkX graph to answer the following questions:

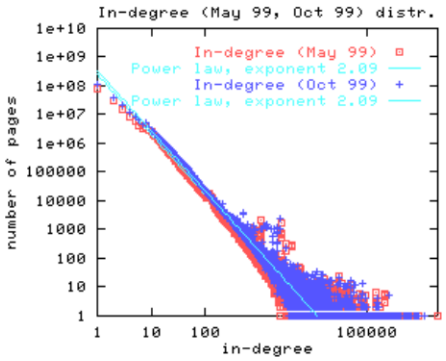

Fig. 3.11 Histogram of Web in-degree on a log-log scale. Reprinted from Broder et al. (2000) with permission from Elsevier.

1 What is the average number of routes served by each airport in this network?
2 What are the top five airports in terms of number of routes?
3 How many airports in this network serve only a single route?
4 Which airport has the highest closeness centrality?
5 Which airport has the highest betweenness centrality?
6 Compute the heterogeneity parameter of this network.

3.18 Load the Wikipedia mathematics network into a NetworkX DiGraph in order to answer the following questions:

1 Compute the average in-degree and average out-degree of this network. What do you notice? Why?
2 Which node has the highest in-degree?
3 Which node has the highest out-degree?
4 In this graph, which is greater: the maximum in-degree or the maximum out-degree? Would you expect this to be the same for other Web graphs? Why?
5 Compute the heterogeneity parameter for this graph's in-degree distribution.
6 Compute the heterogeneity parameter for this graph's out-degree distribution.

3.19 Write a Python function that accepts a NetworkX graph and a node name and returns the average degree of that node's neighbors. Use this function to compute this quantity for every node in the OpenFlights US network and take the average. Does the friendship paradox hold here, *i.e.*, is the average degree of nearest neighbors greater than the average node degree?

3.20 Are there networks such that the average number of neighbors of a node's neighbors match the average degree? If there are, what property must they have?

3.21 Are networks with heavy-tailed degree distributions more vulnerable to random or targeted attacks? And what about grid-like networks of similar size?

3.22 If one seeks to disrupt a network by removing nodes and/or edges in an effort to disconnect it and/or increase the average path length, an obvious strategy is to attack the hubs. Which of the following is another deleterious criterion for selecting targets? Explain your answer.

a. Nodes with high clustering coefficient
b. Nodes with low degree
c. Nodes with high closeness centrality
d. Nodes/edges with high betweenness centrality

3.23 Consider two nodes of equal degree on some network: one with high clustering coefficient and one with low clustering coefficient. All else being equal, which of the two would you intuit to be a better target if you were seeking to disrupt the network?

3.24 The `socfb-Northwestern25` network in the book's Github repository is a snapshot of Northwestern University's Facebook network. The nodes are anonymous users and the links are friend relationships. Load this network into a NetworkX graph in order to answer the following questions. Be sure to use the proper graph class for an undirected, unweighted network.

1 What proportion of nodes have degree 100 or greater?
2 What is the maximum degree for nodes in this network?
3 Users in this network are anonymized by giving the nodes numerical names. Which node has the highest degree?
4 What is 95th percentile for degree, *i.e.*, the value such that 95% of nodes have this degree or less?
5 What is the mean degree for nodes in this network? Round to the nearest integer.
6 Which of the following shapes best describes the degree distribution in this network? You can obtain the answer visually using histograms, or just with statistics.

a. Uniform: node degrees are evenly distributed between the minimum and maximum.
b. Normal: most node degrees are near the mean, dropping off rapidly in both directions.
c. Right-tailed: most node degrees are relatively small compared to the range of degrees.
d. Left-tailed: most node degrees are relatively large compared to the range of degrees.