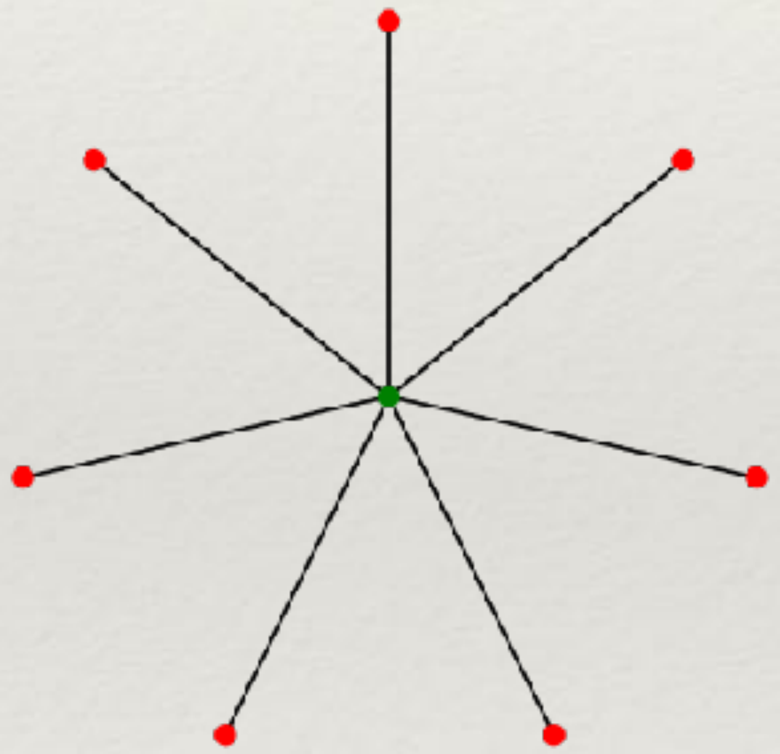
Basics of CNA: Complex Network Analysis

Overview of CNA

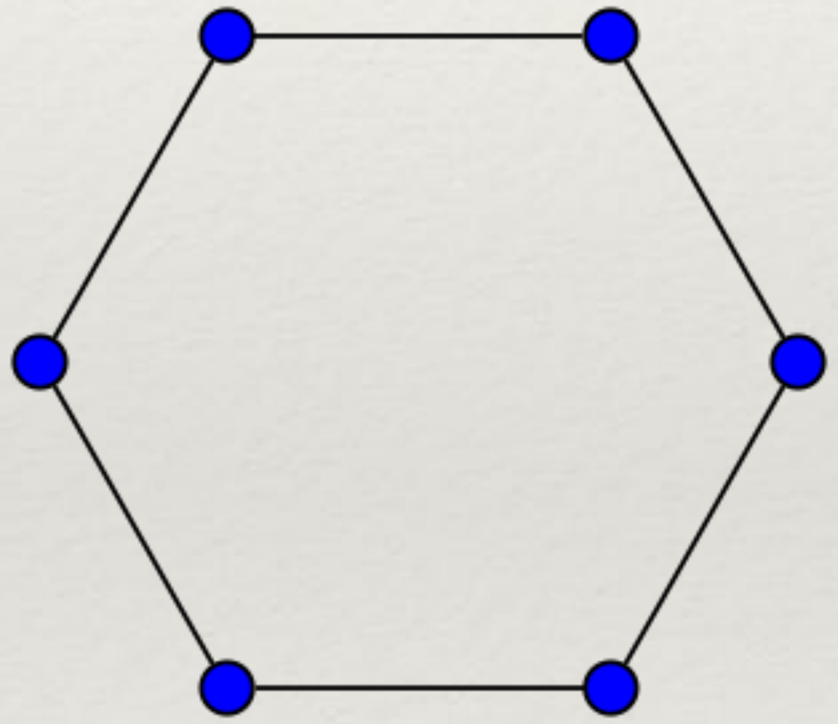
- different roles: hubs, weak ties, bridges, betweenness
- network heterogeneity
- robustness and immunization
- weighted and directed networks
- communities
- homophily
- the emergence of social clusters and segregation
- information and misinformation

Networks structural aspects

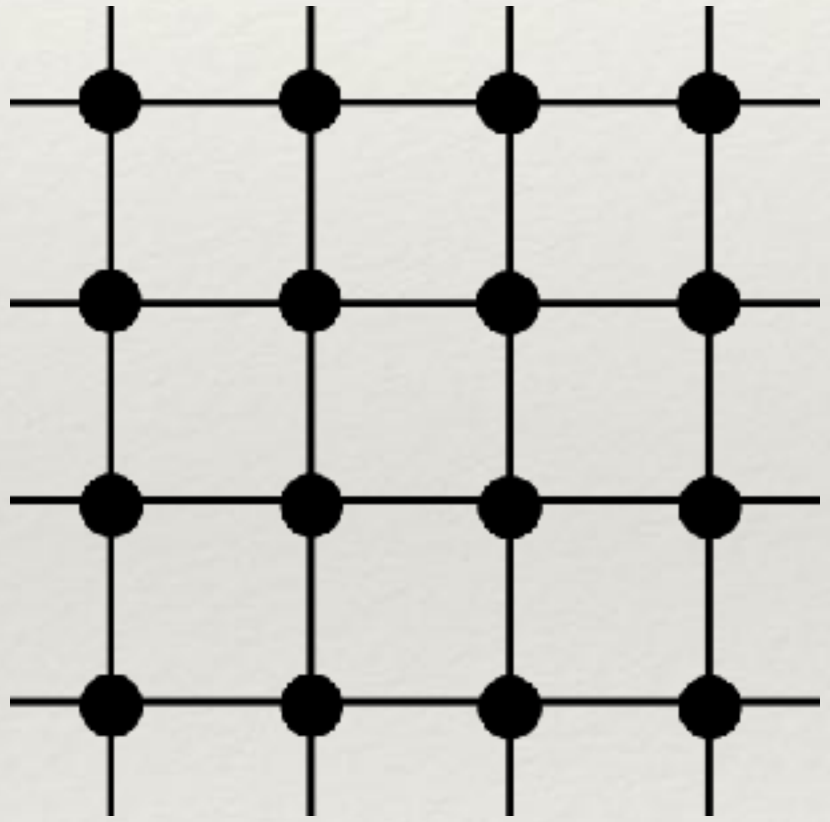
- "trivial" **representation** of a complex system
- **Simple** networks: few characteristics describe the network



Star



Ring



Grid

- We need a **language** and a **framework** to describe complex networks

Basic Definitions

- A **graph** (or a network) is made of nodes and links
- **nodes** (or vertices)
- **links** (or edges, or arcs)
- Graphs can be **directed** or **undirected**
- Graphs can be **weighted** or **unweighted**

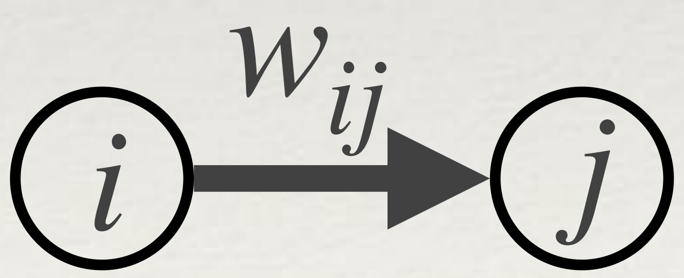
$$G = (N, L)$$

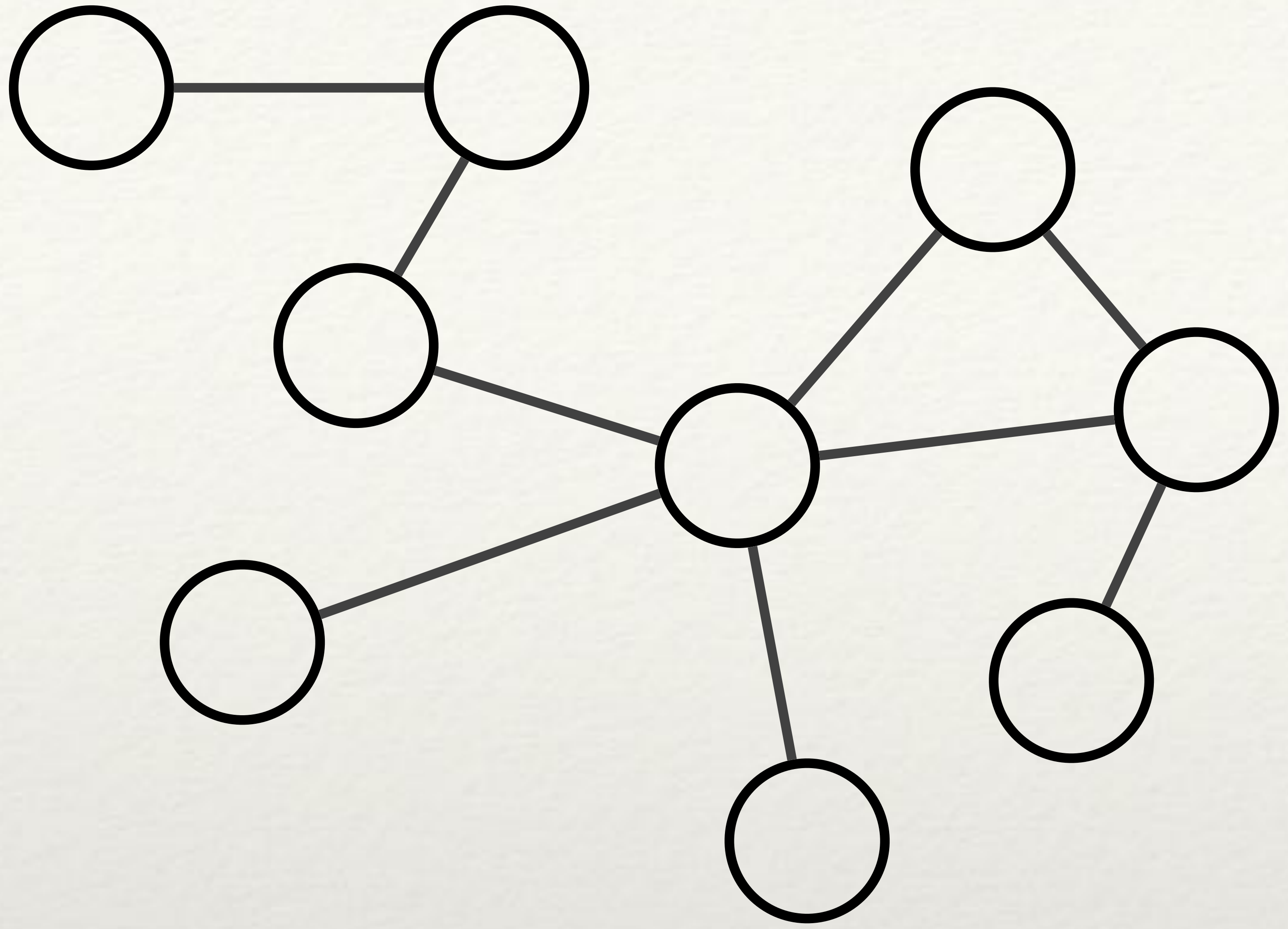
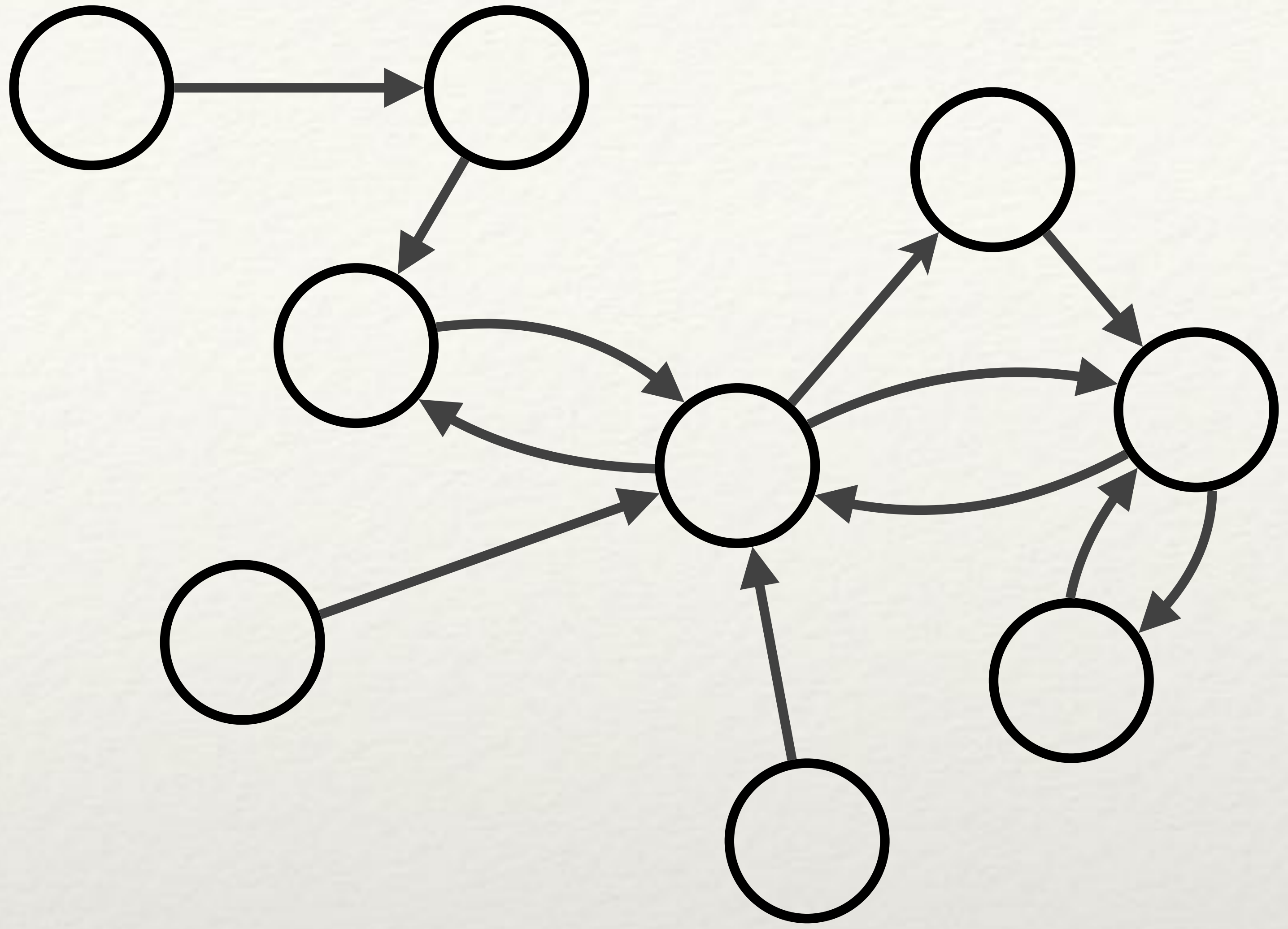
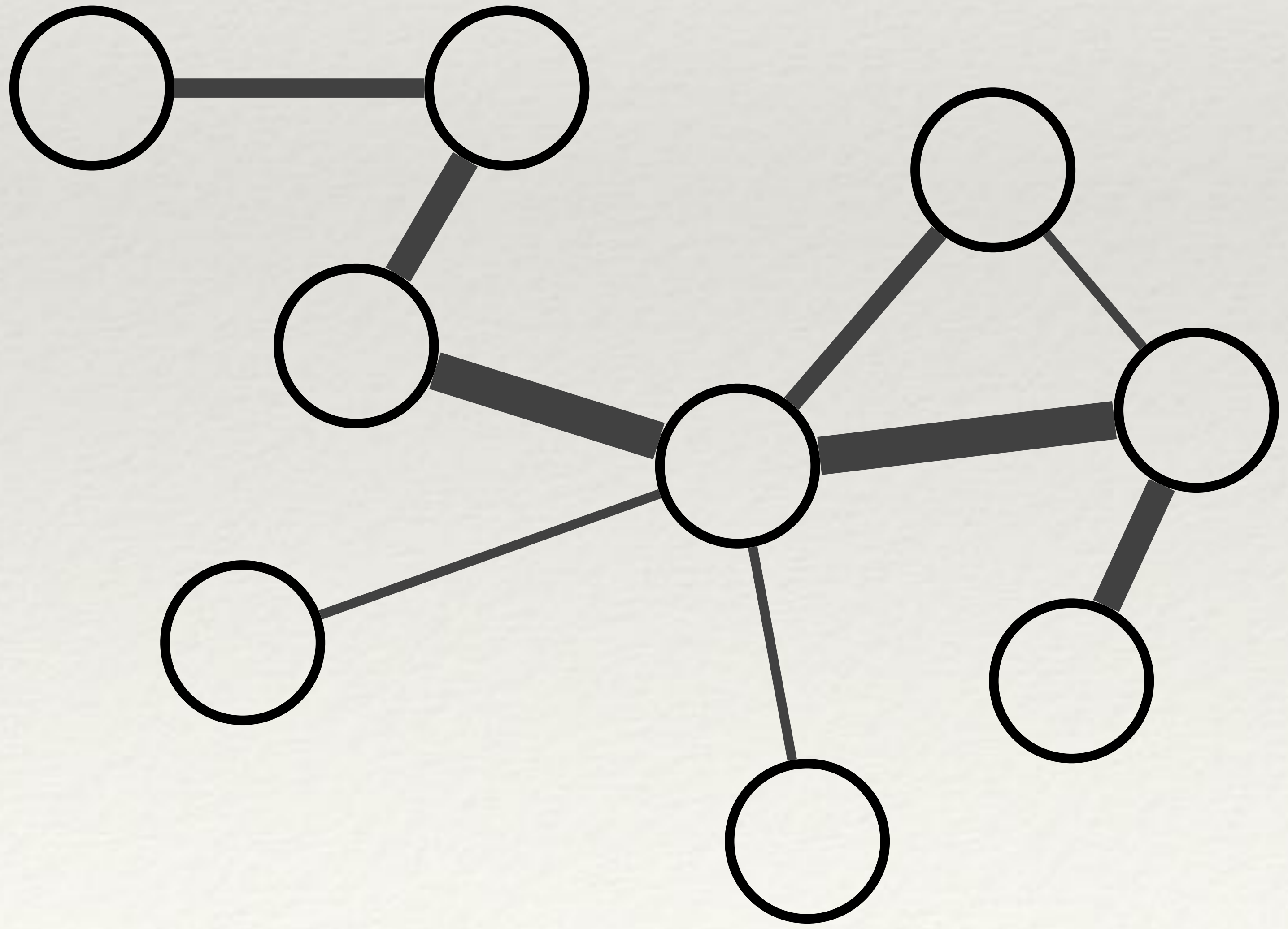
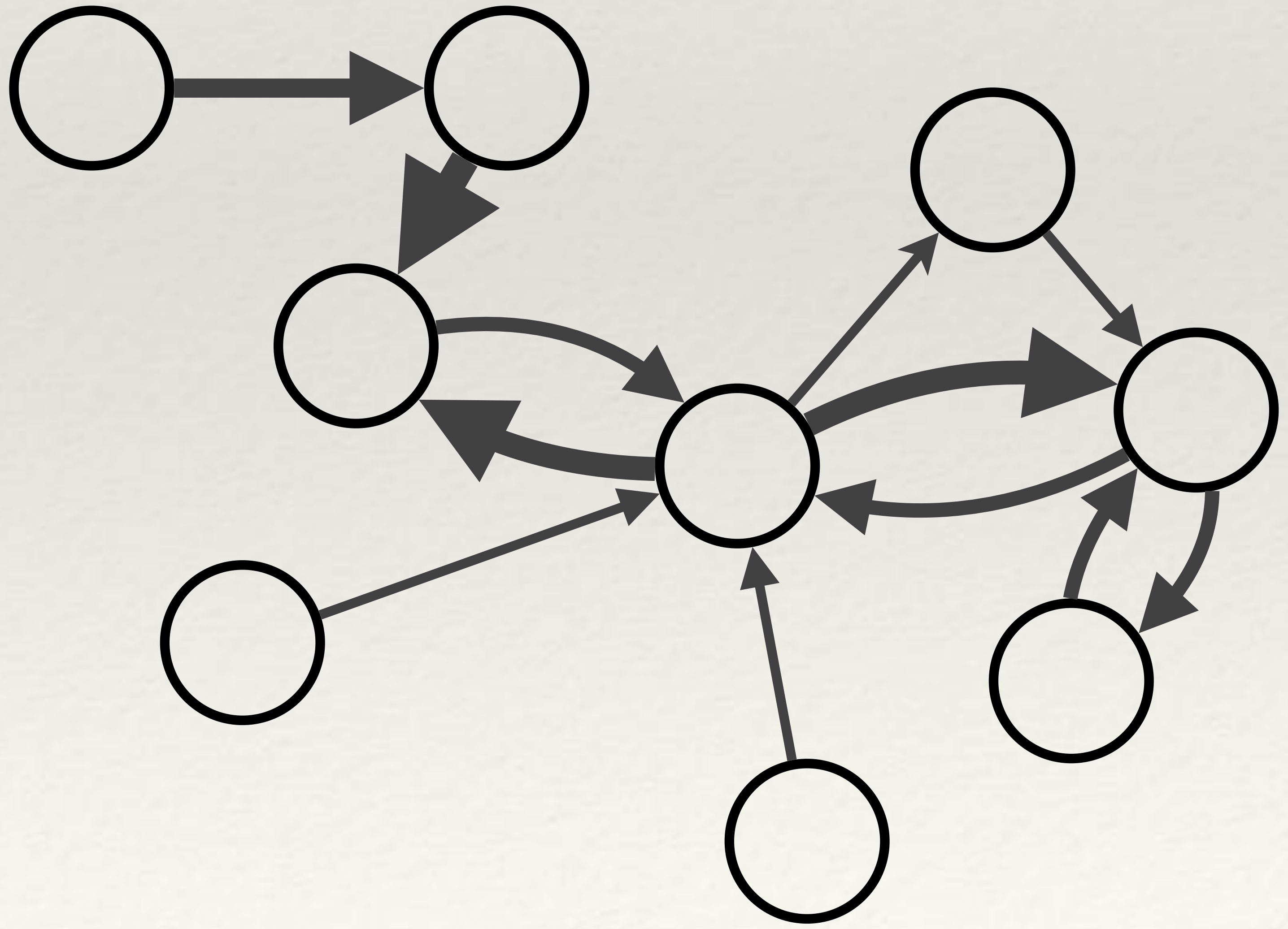
$$N = \{n_1, n_2, \ldots, n_l\} = \{1,2,\ldots,l\}$$

$$L = \{(i,j) : i,j \in N\}$$

(i,j)

w_{ij} (i,j,w)



	Undirected	Directed
Unweighted		
Weighted		

Degree

Number of links (or neighbors)

$i \to N_i \qquad k_i = |N_i|$ degree

Singleton: a node whose degree is zero

$N_i = \{\}, k_i = 0$

In directed networks

$k_i^{in} = |P_i|$ in-degree

$k_i^{out} = |S_i|$ out-degree

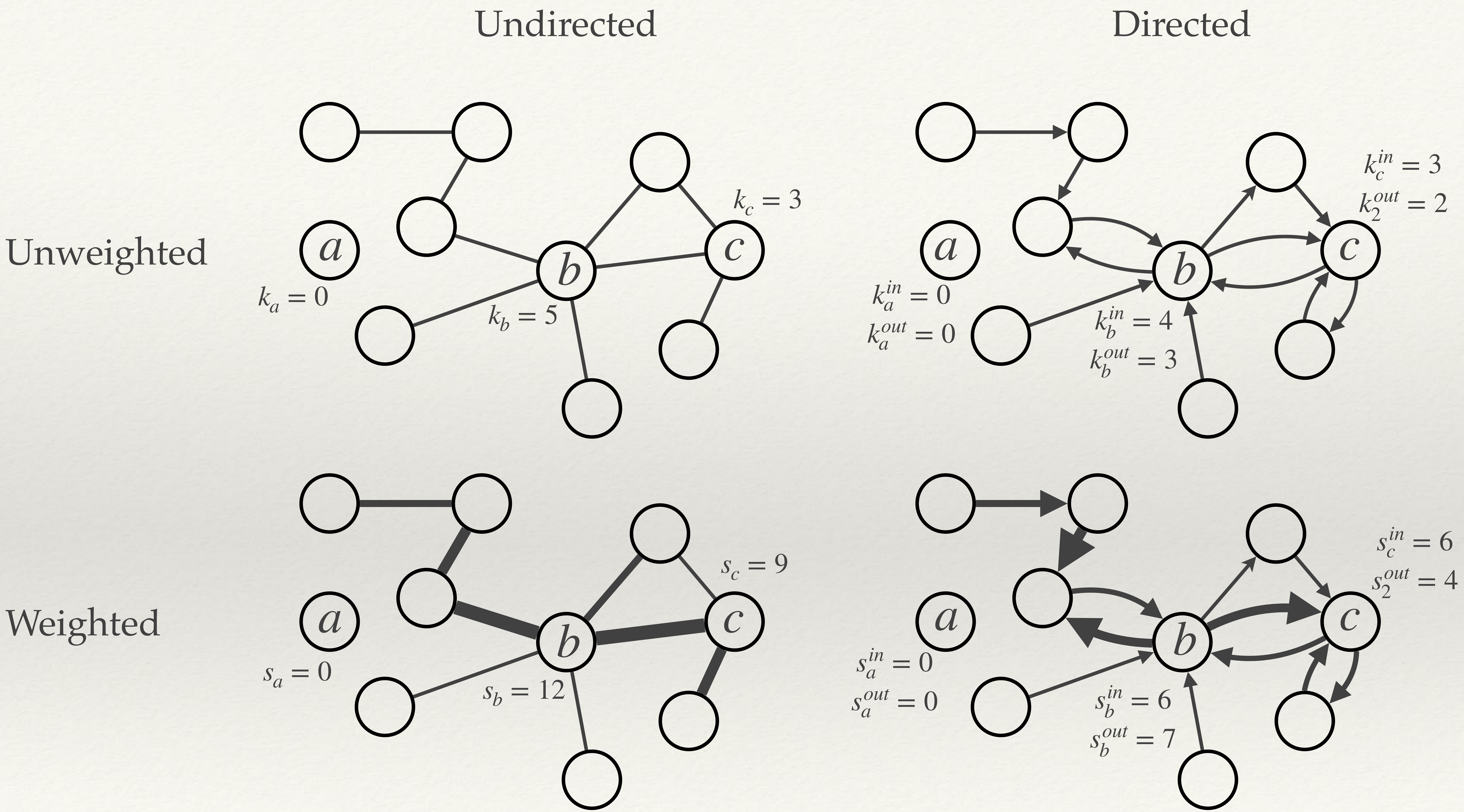
$$k_i = k_i^{in} + k_i^{out}$$

Strength

Strength: Weighted degree $$s_i = \sum_{j \in N_i} w_{ij}$$

in-strength $$s_i^{in} = \sum_{j \in P_i} w_{ji}$$

out-strength $$s_i^{out} = \sum_{j \in S_i} w_{ij}$$


Undirected
Directed
Unweighted
Weighted
a
b
c
$k_a = 0$
$k_b = 5$
$k_c = 3$
$k_a^{in} = 0$
$k_a^{out} = 0$
$k_b^{in} = 4$
$k_b^{out} = 3$
$k_c^{in} = 3$
$k_2^{out} = 2$
$s_a = 0$
$s_b = 12$
$s_c = 9$
$s_a^{in} = 0$
$s_a^{out} = 0$
$s_b^{in} = 6$
$s_b^{out} = 7$
$s_c^{in} = 6$
$s_2^{out} = 4$

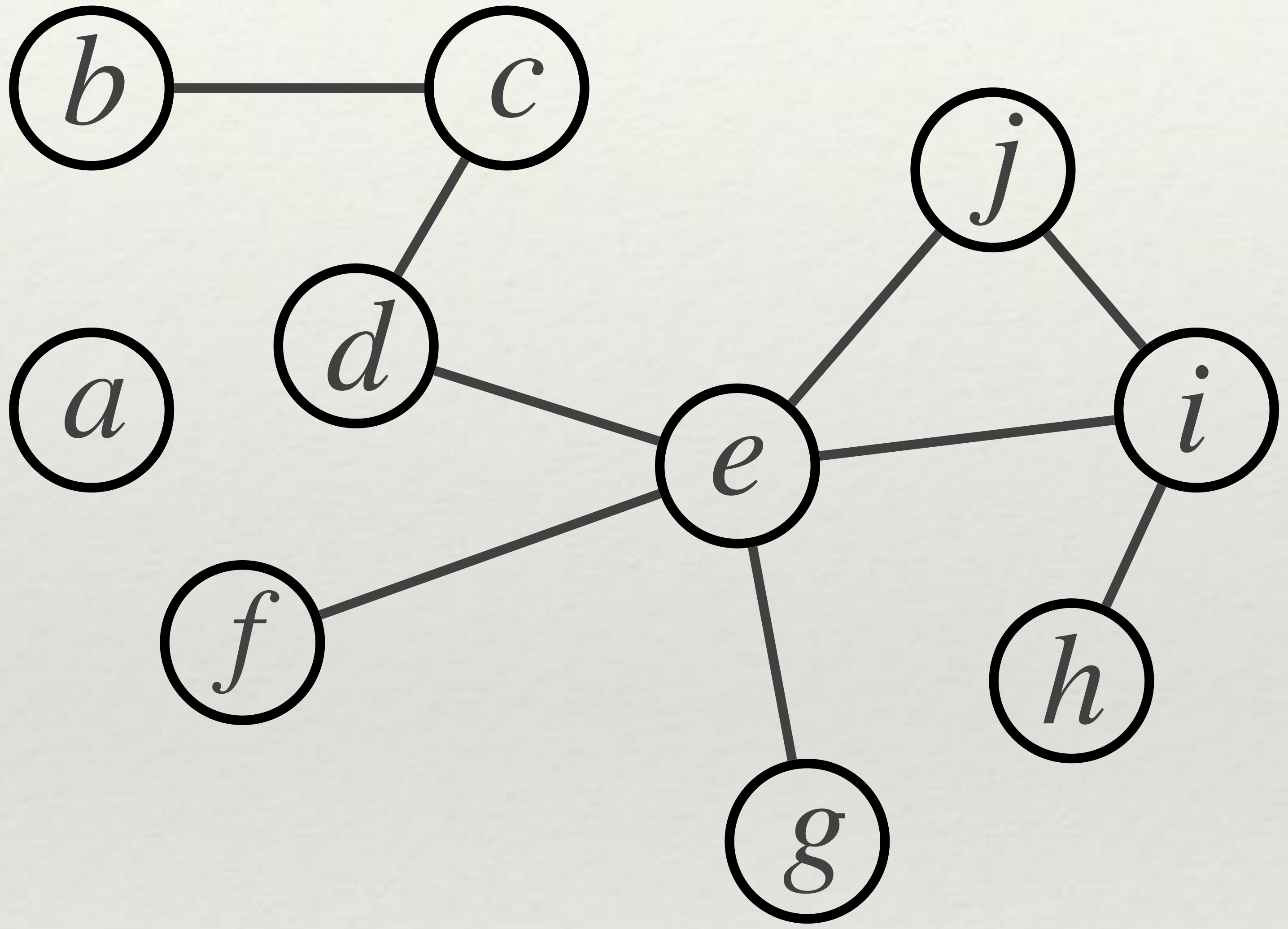
Network representations

Adjacency Matrix

NxN matrix

$$a_{ij} = \begin{cases} 0 & \text{no edge} \\ 1 & (i,j) \in L \end{cases}$$

Undirected network: $a_{ij} = a_{ji}$



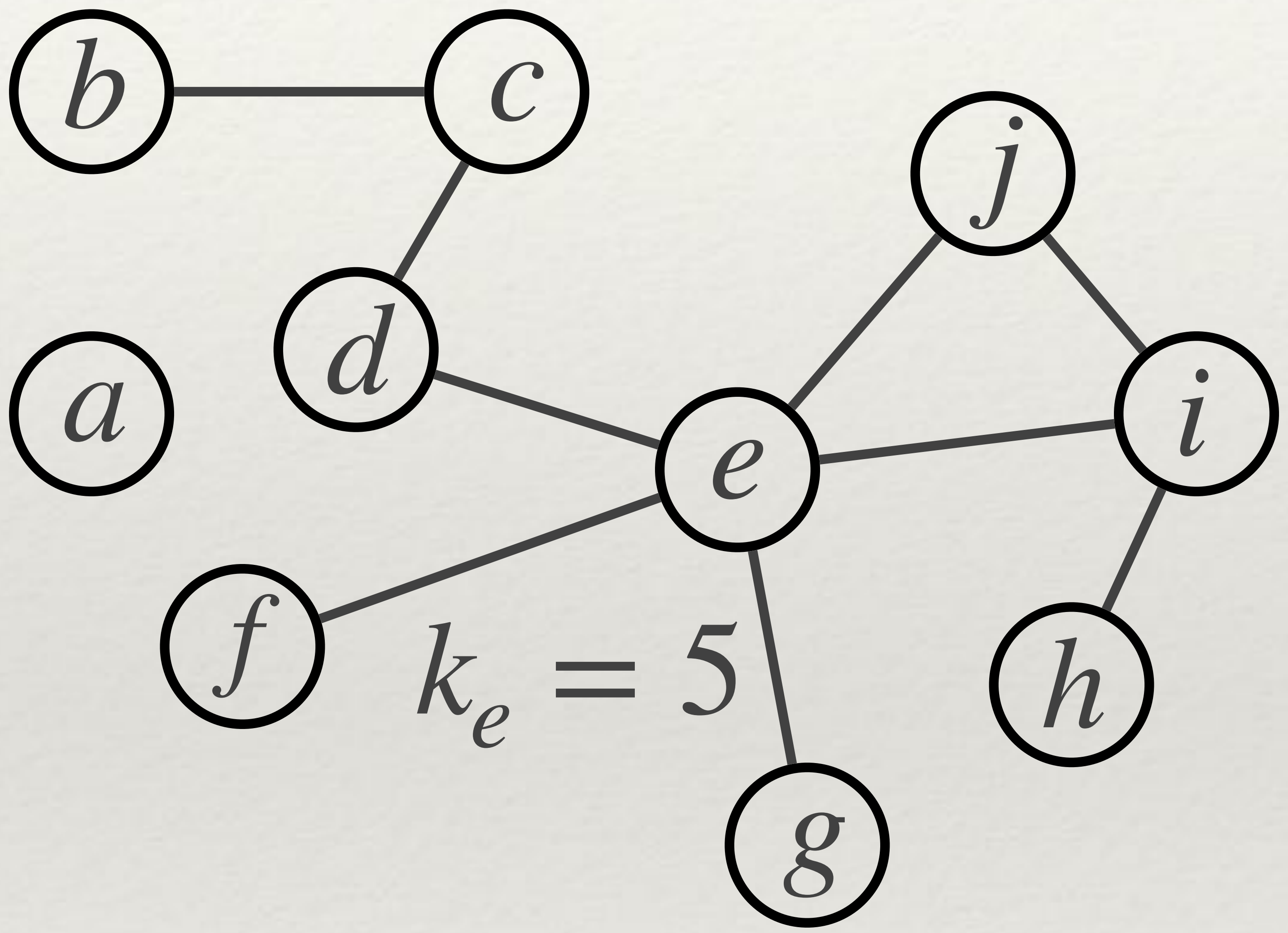
$$\begin{array}{rc}
 & \begin{array}{cccccccccc} a & b & c & d & e & f & g & h & i & j \end{array} \\
\begin{array}{c} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \\ j \end{array} &
\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0
\end{pmatrix}
\end{array}$$

Network representations

Adjacency Matrix

degree

$$k_i = \sum_j a_{ij} = \sum_j a_{ji}$$



$$
\begin{array}{c|c}
 & \begin{array}{cccccccccc} \mathtt{a} & \mathtt{b} & \mathtt{c} & \mathtt{d} & \mathtt{e} & \mathtt{f} & \mathtt{g} & \mathtt{h} & \mathtt{i} & \mathtt{j} \end{array} \\
\begin{array}{c} \mathtt{a} \\ \mathtt{b} \\ \mathtt{c} \\ \mathtt{d} \\ \mathtt{e} \\ \mathtt{f} \\ \mathtt{g} \\ \mathtt{h} \\ \mathtt{i} \\ \mathtt{j} \end{array} &
\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0
\end{pmatrix}
\end{array}
$$

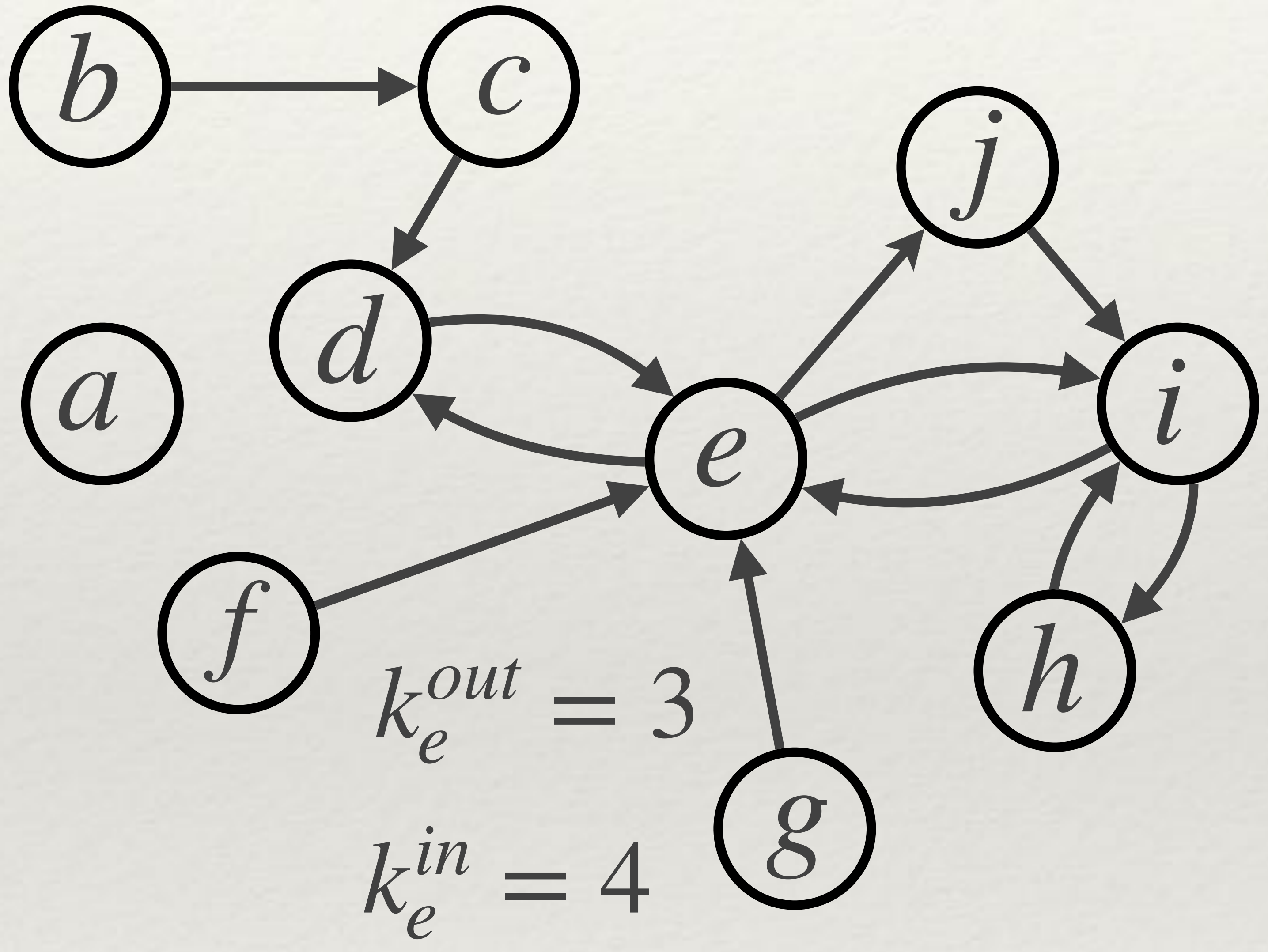
Network representations

Adjacency Matrix
directed

Matrix is not symmetric

$$k_i^{out} = \sum_j a_{ij}$$

$$k_i^{in} = \sum_j a_{ji}$$



$$\begin{array}{c c}
 & \begin{array}{cccccccccc} a & b & c & d & e & f & g & h & i & j \end{array} \\
\begin{array}{c} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \\ j \end{array} &
\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{pmatrix}
\end{array}$$

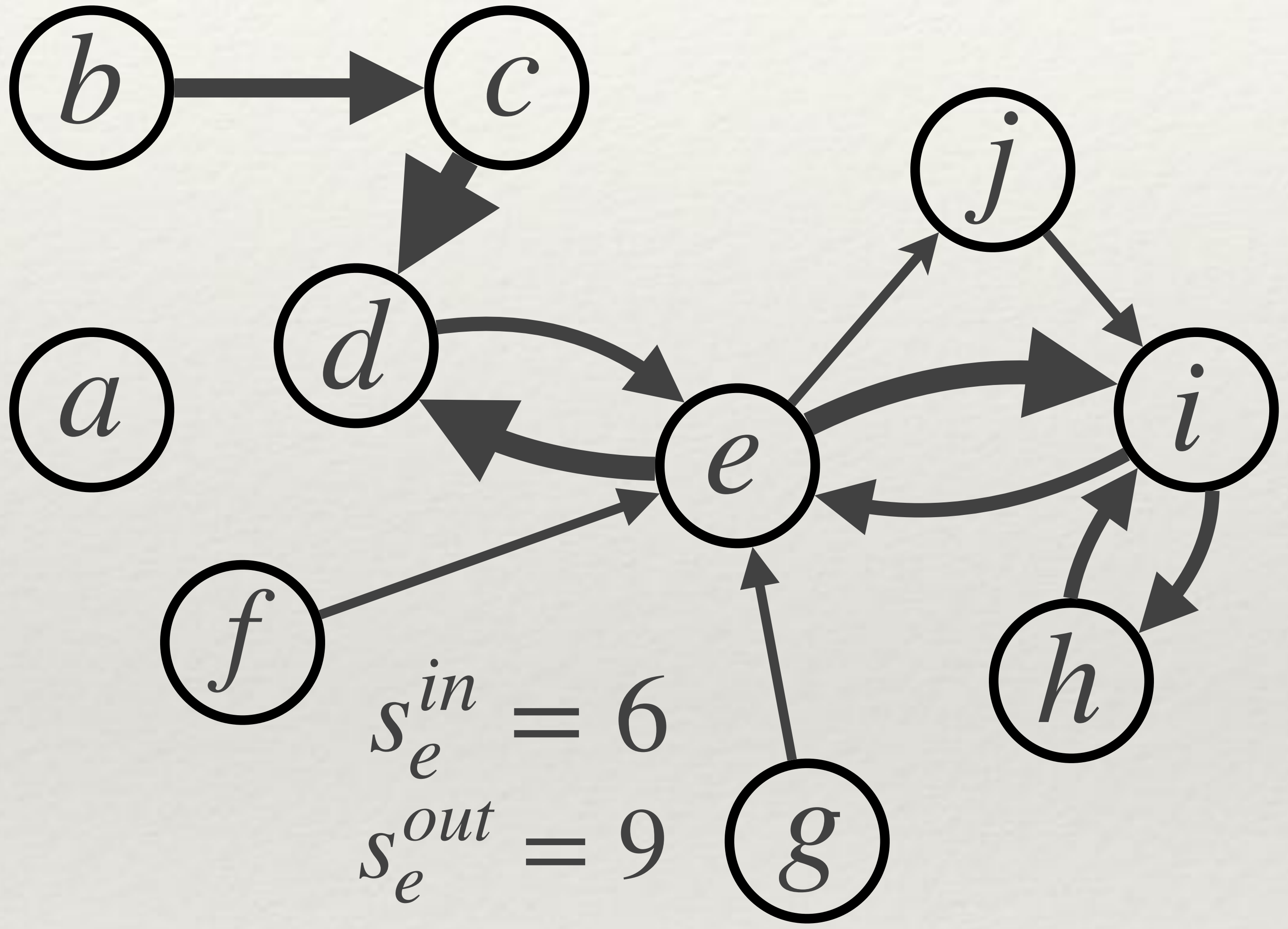
Network representations

Adjacency Matrix
weighted

$$w_{ij}$$

$$s_i^{out} = \sum_j w_{ij}$$

$$s_i^{in} = \sum_j w_{ji}$$

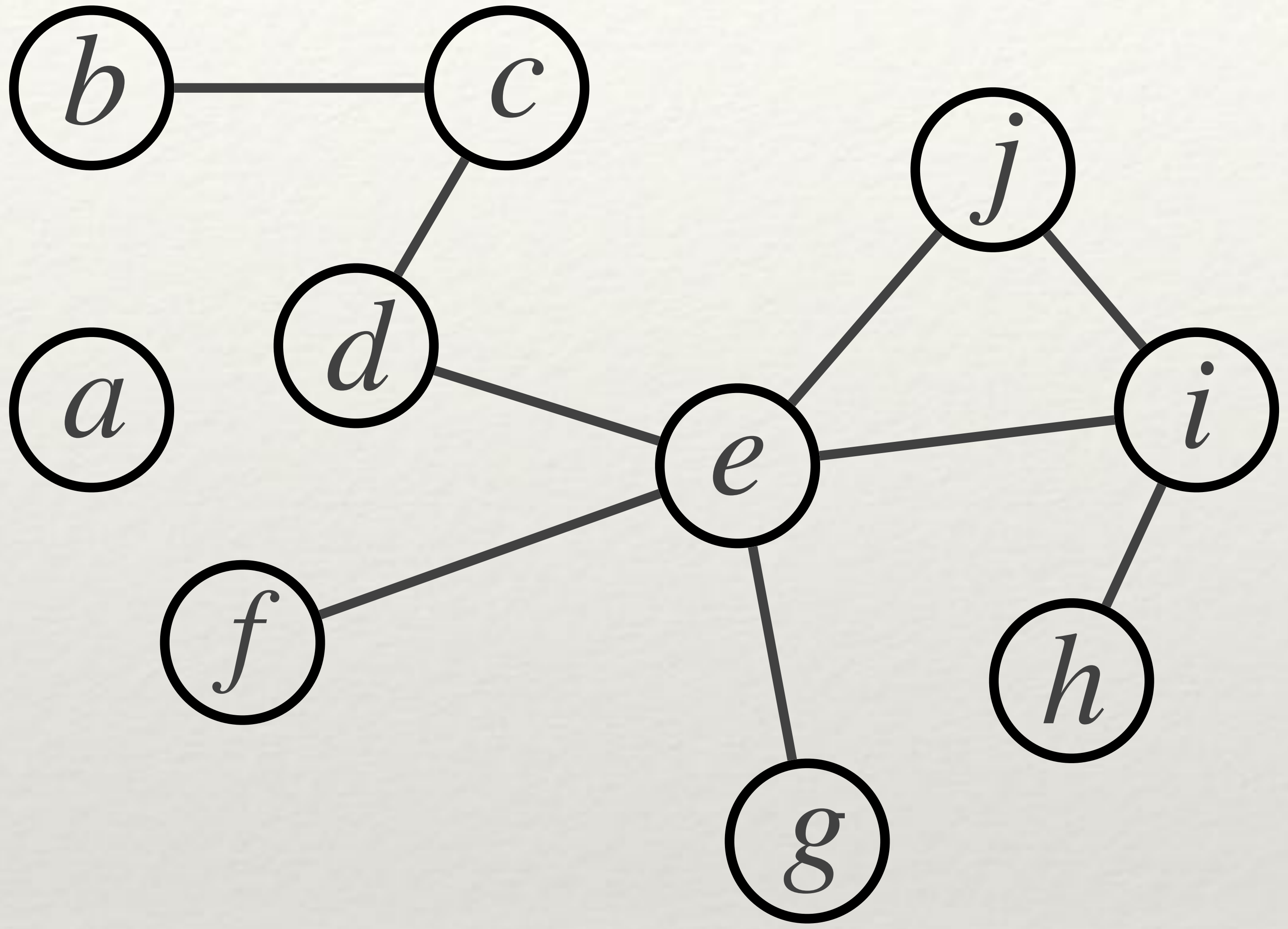


$$\begin{array}{c c} & \begin{array}{cccccccccc} a & b & c & d & e & f & g & h & i & j \end{array} \\ \begin{array}{c} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \\ j \end{array} & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 4 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \end{pmatrix} \end{array}$$

Sparse network representations

- The memory/disk storage needed by an adjacency matrix is proportional to N^2
- In sparse networks (most real-world networks), this is terribly inefficient: most of the space is wasted storing zeros (non-links); for very large networks, adjacency matrices are unfeasible
- It is much more efficient, often necessary, to store only the actual links, and assume that if a link is not listed it means it is not present
- There are two commonly used sparse networks representations:
  - **Adjacency list**
  - **Edge list**

Adjacency list

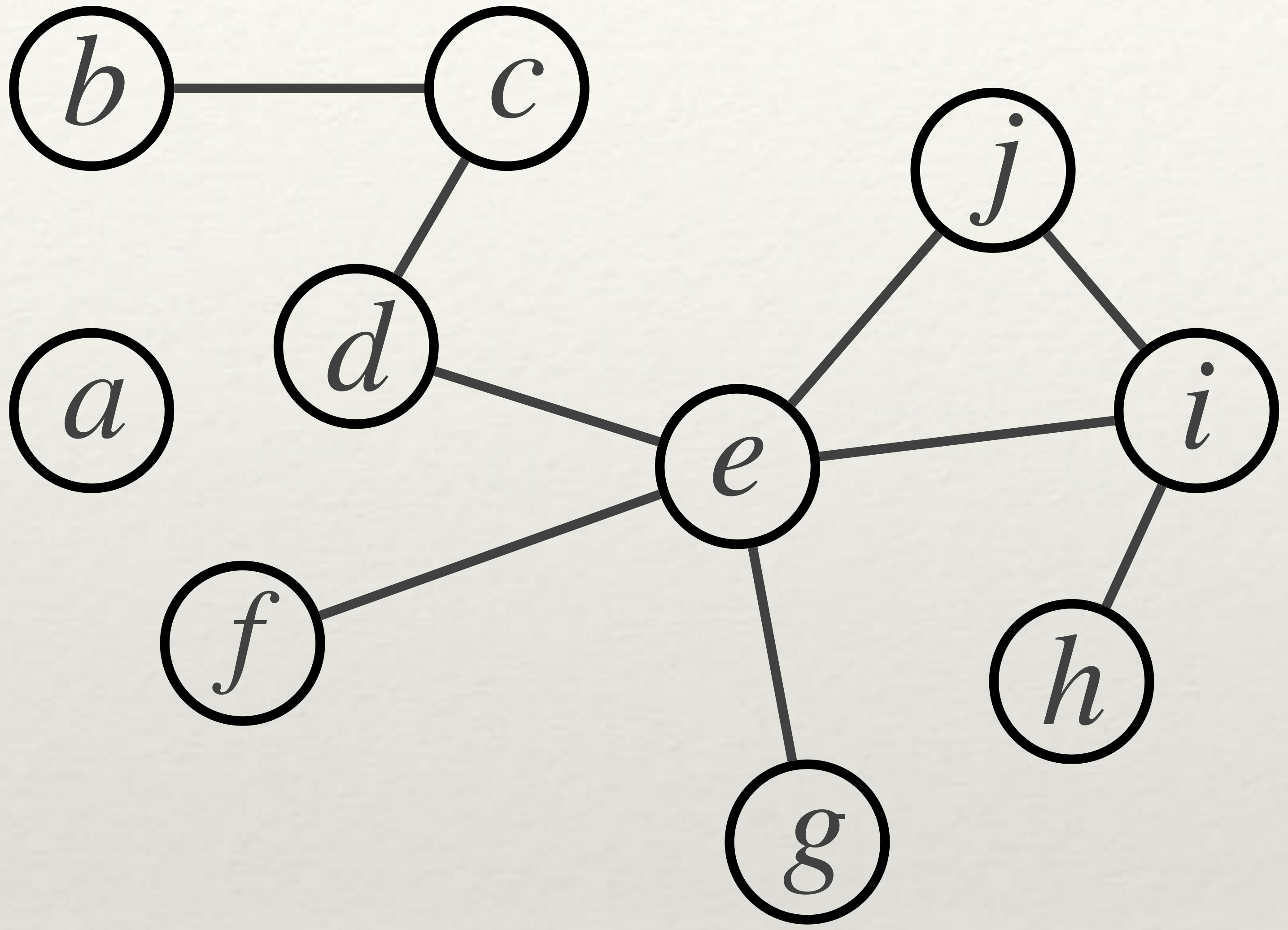



```
a
b    c
c    b d
d    c e
e    d f i j g
f    e
g    e
h    i
i    h j
j    e i
```

Undirected network: list each link twice

Directed network: list only existing links

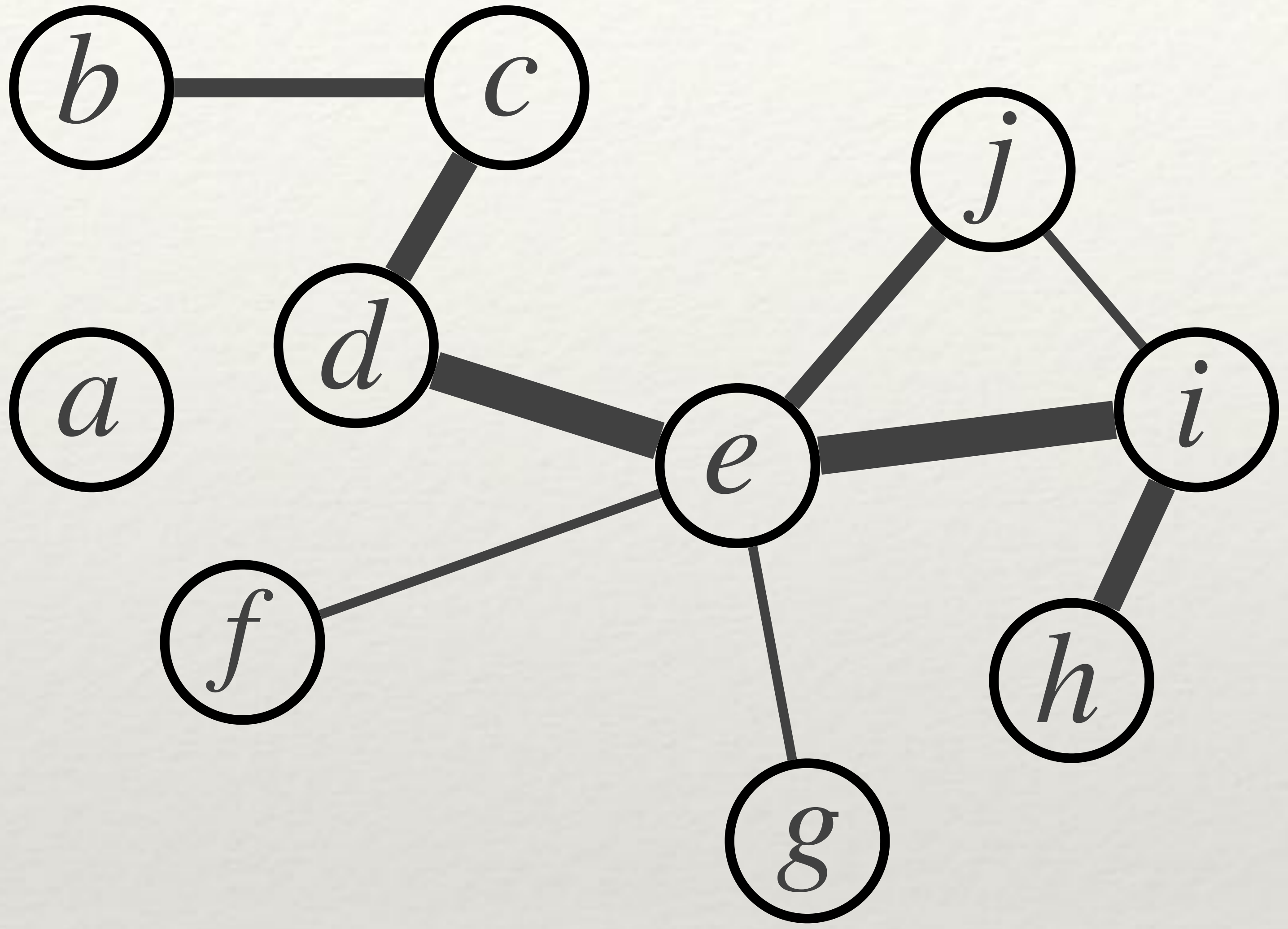
Edge list



b	c
c	d
d	e
e	f
e	g
e	i
e	j
h	i
i	j

L

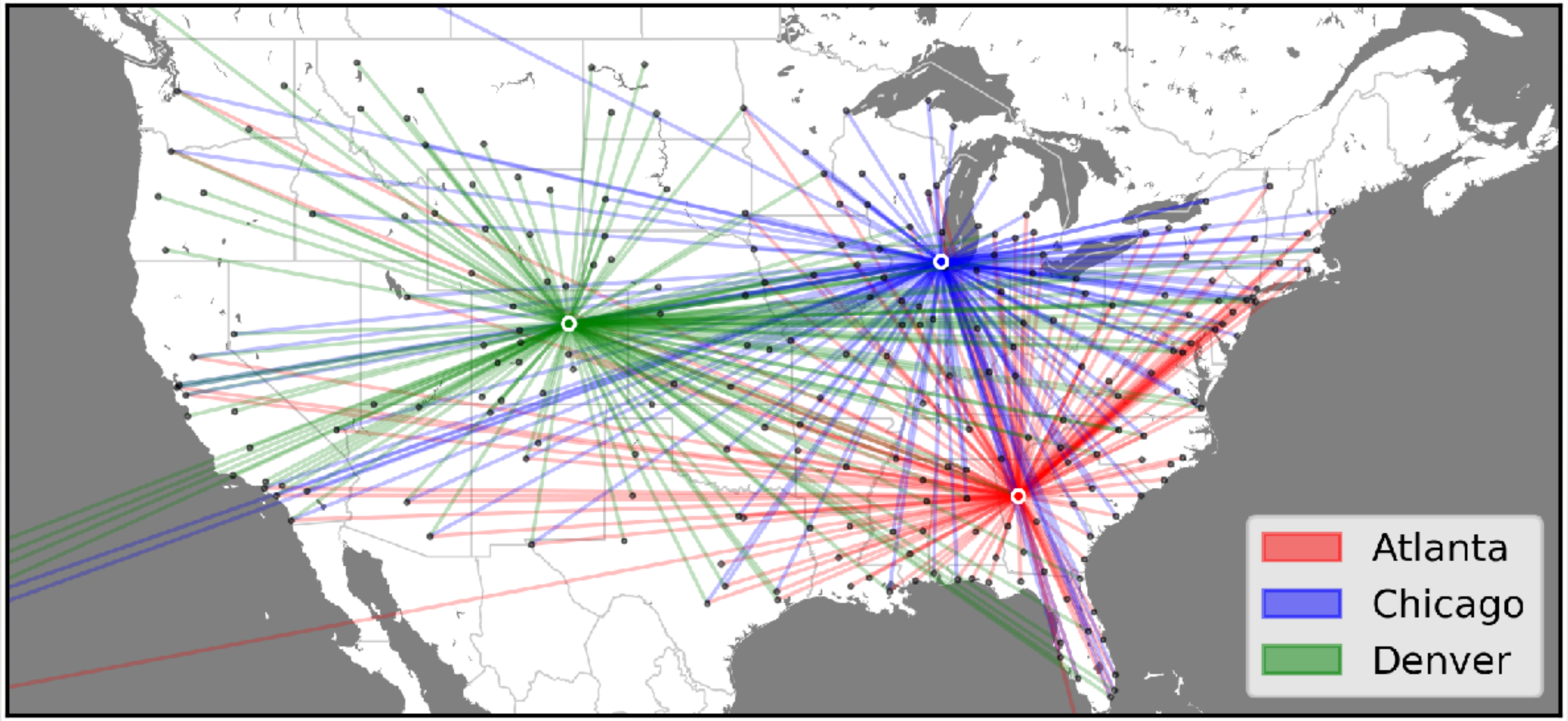
Edge (weighted) list



b	c	2
c	d	3
d	e	4
e	f	4
e	g	1
e	i	1
e	j	2
h	i	3
i	j	1

L

Real networks are heterogeneous



Some nodes (and links) are much more important (**central**) than others!

A First Course in Network Science by F. Menczer, S. Fortunato & C.A. Davis. Cambridge University Press, 2020

Centrality measures

- **Centrality**: measure of importance of a node
- **Measures**:

1. Degree
2. Closeness
3. Betweenness

Degree

- **Degree of a node:** number of neighbors of the node

$$k_i = \text{number of neighbors of node } i$$

- High-degree nodes are called **hubs**

- **Average degree of the network:**

$$\langle k \rangle = \frac{\sum_i k_i}{N} = \frac{2L}{N}$$

```
G.degree(2) # returns the degree of node 2
G.degree()  # dict with the degree of all nodes of G
```

Closeness

Idea: a node is the more central the *closer* it is to the other nodes, on average

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

where ℓ_{ij} is the distance between nodes *i* and *j*

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

Betweenness

Idea: a node is the more central the *more often it is crossed by paths*

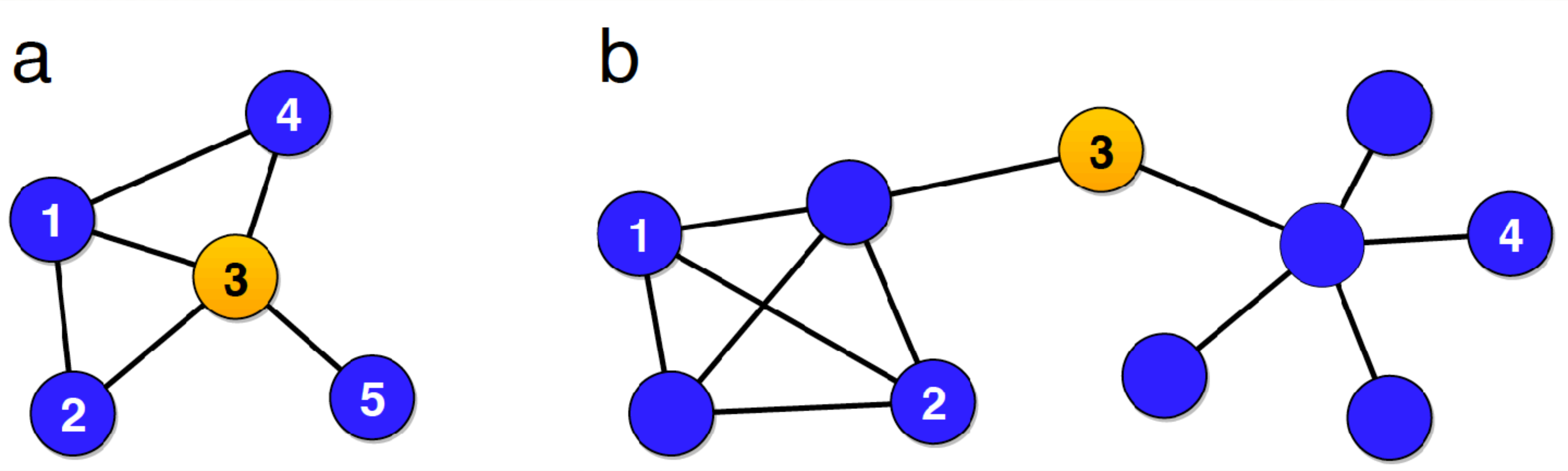
$$b_i = \sum_{h \neq j \neq i} \frac{\sigma_{hj}(i)}{\sigma_{hj}}$$

σ_{hj} = **number of shortest paths from h to j**

$\sigma_{hj}(i)$ = **number of shortest paths from h to j running through i**

Betweenness

Hubs usually have high betweenness, but there can be nodes with high betweenness **that are not hubs**



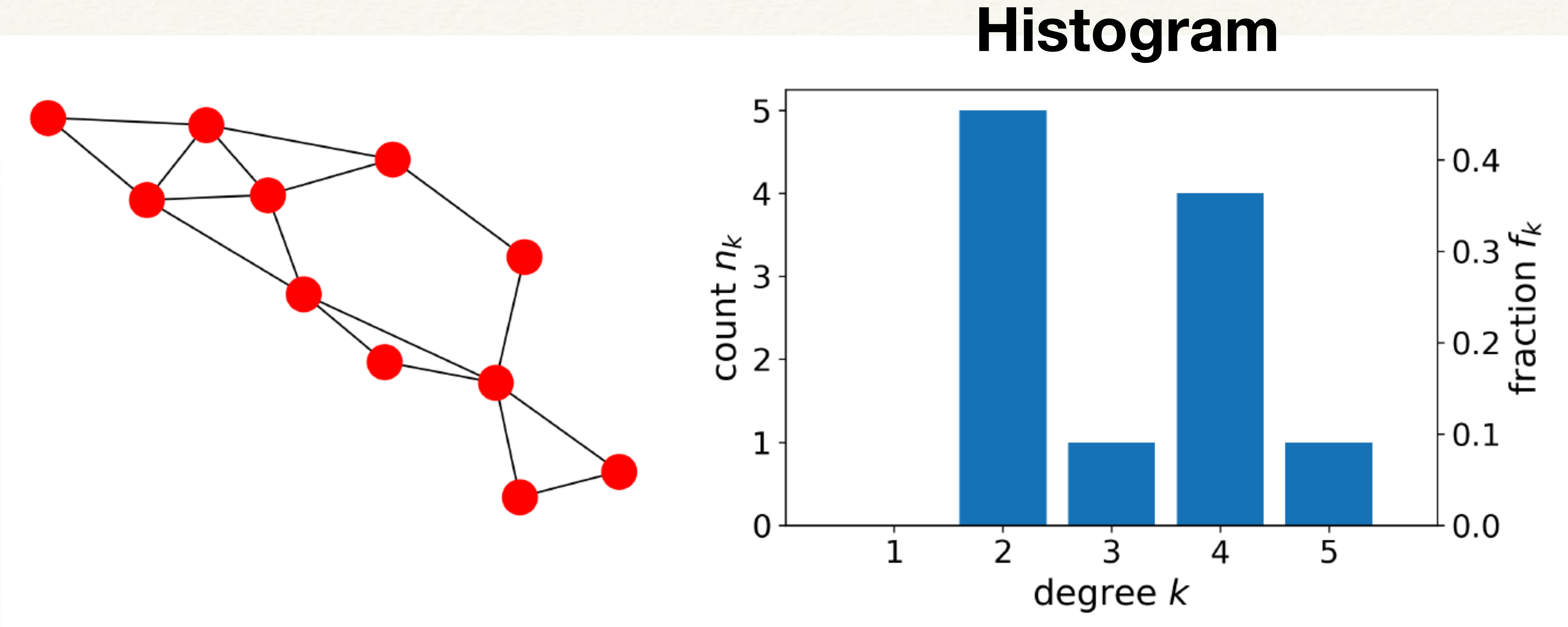
Betweenness

- Betweenness can be easily extended to links
- **Link betweenness:** fraction of shortest paths among all possible node pairs that pass through the link

Centrality distributions

- On small networks it makes sense to ask which nodes or links are most important
- On large networks **it does not**
- **Solution:** statistical approach
- Instead of focusing on individual nodes and links, we consider **classes** of nodes and links with similar properties

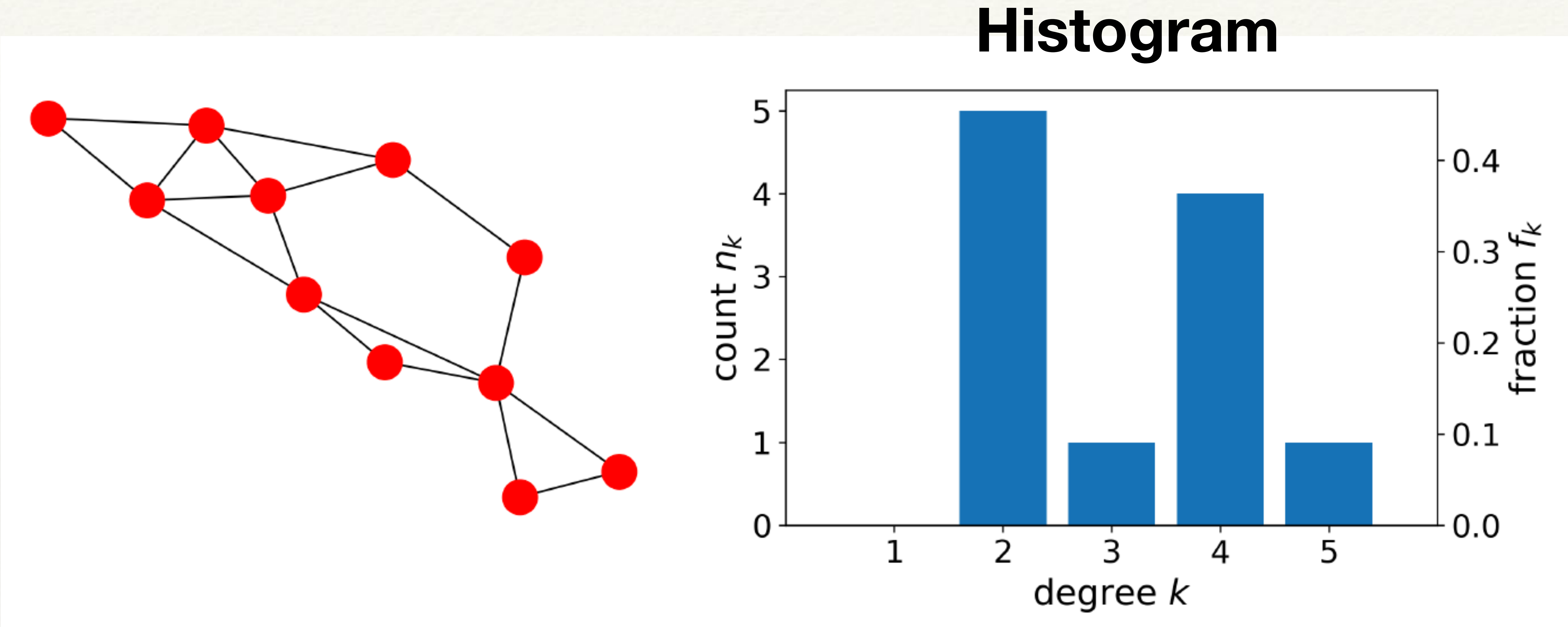
Centrality distributions



n_k = number of nodes with degree k

$f_k = \frac{n_k}{N}$ = frequency of degree k

Centrality distributions



- For large N, the frequency f_k becomes the **probability** p_k of having degree k
- **Probability distribution:** plot of probability p_k versus k

Cumulative distributions

- If the variable is *not integer* (e.g., betweenness), the range of the variable is divided into intervals (bins) and we count how many values fall in each interval

- **Cumulative distribution *P(x)*:** probability that the variable takes values *larger* than *x* as a function of *x*

- **How to compute it:** by summing the frequencies of the variable inside the intervals to the right of *x*

$$P(x) = \sum_{v \geq x} f_v$$

Logarithmic scale

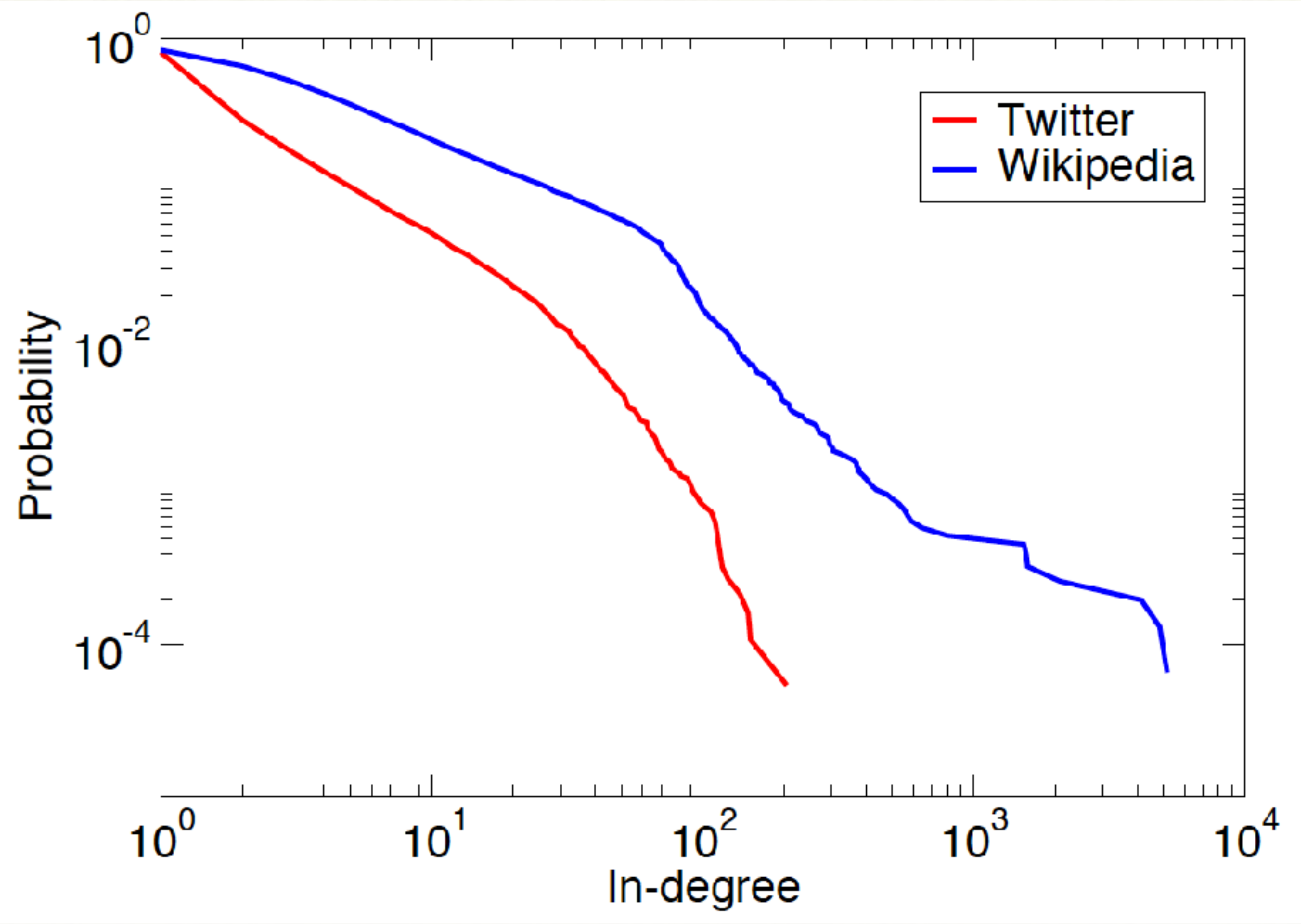
- **Question:** how to plot a probability distribution if the variable spans a large range of values, from small to (very) large?
- **Answer:** use the **logarithmic** scale
- **How to do it:** report the logarithms of the values on the x- and y-axes

$$\log_{10} 10 = 1$$

$$\log_{10} 1{,}000 = \log_{10} 10^3 = 3$$

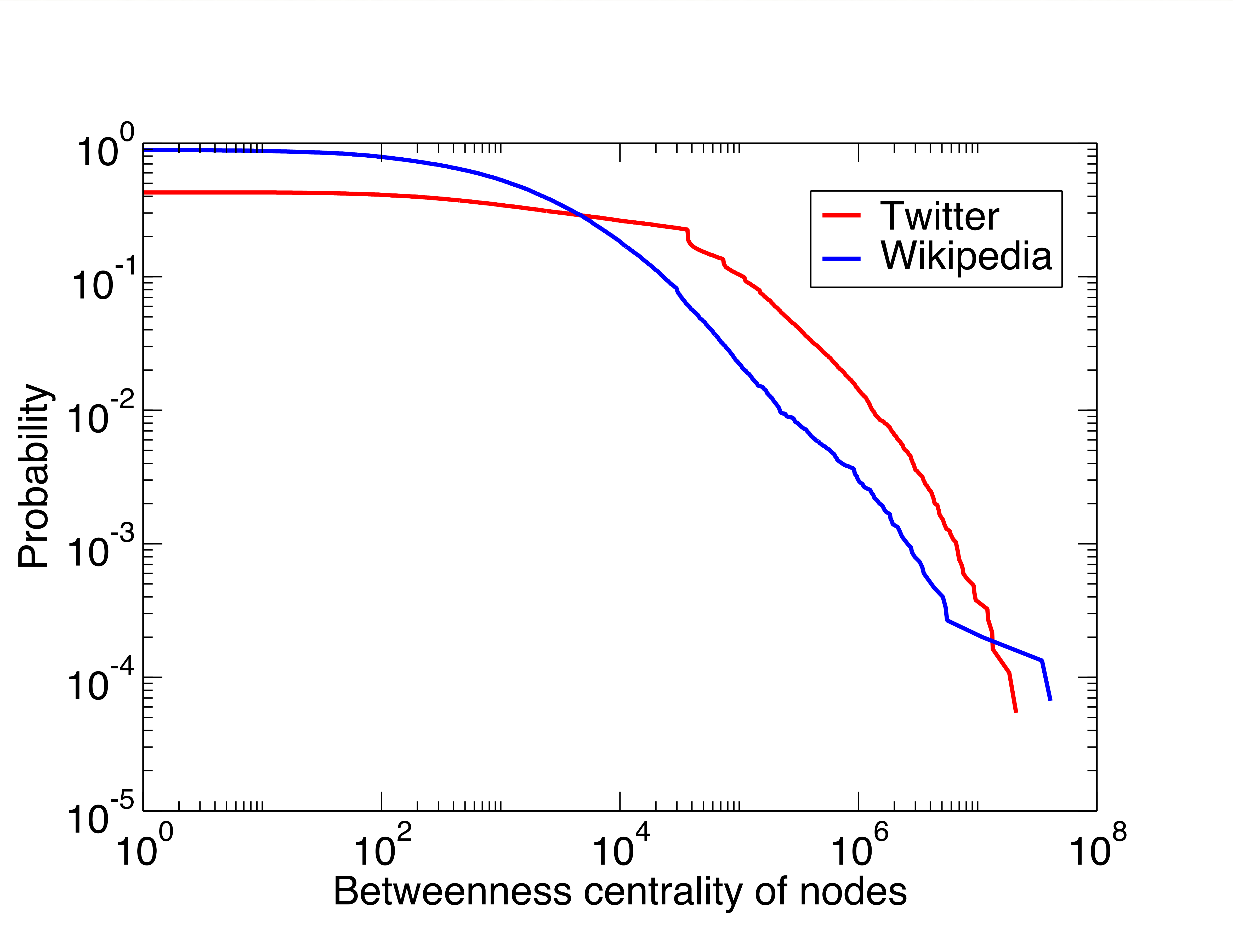
$$\log_{10} 1{,}000{,}000 = \log_{10} 10^6 = 6$$

Degree distributions



Heavy-tail distributions: the variable goes from small to large values

Betweenness distributions



Heavy-tail distribution: the variable goes from small to large values

Robustness

- A system is **robust** if the failure of some of its components does not affect its function

- **Question:** how can we define the robustness of a network?

- **Answer:** we remove nodes and/or links and see what happens to its structure

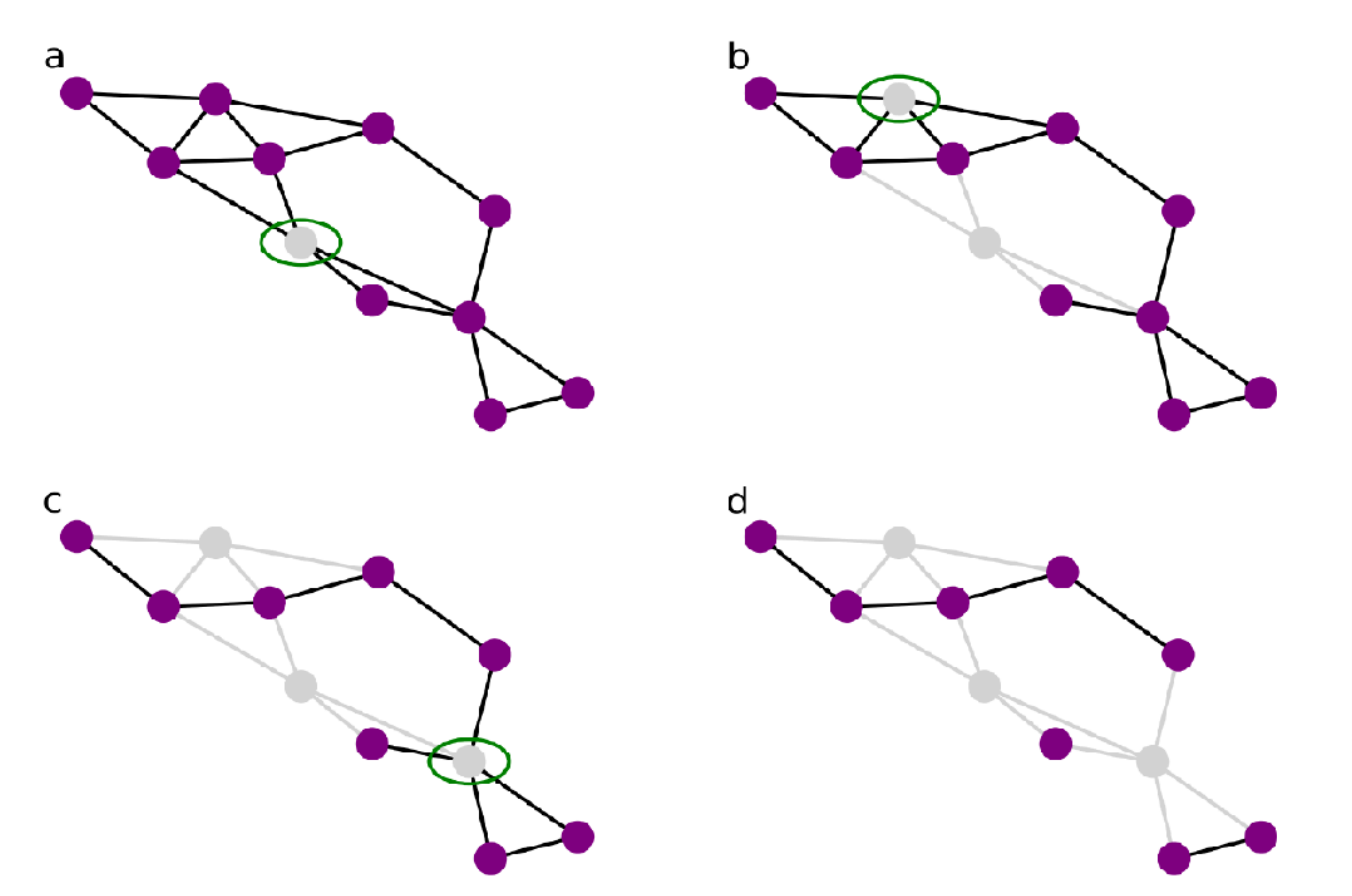
- **Key point:** *connectedness*

- If the Internet were not connected, it would be impossible to transmit signals (e.g., emails) between routers in different components

Robustness

- **Robustness test:** checking how the connectedness of the network is affected as more and more nodes are removed
- **How to do it:** plot the relative size S of the largest connected component as a function of the fraction of removed nodes
- We suppose that the network is initially connected: there is only one component and $S = 1$
- As more and more nodes (and their links) are removed, the network is progressively broken up into components and S goes down

Robustness



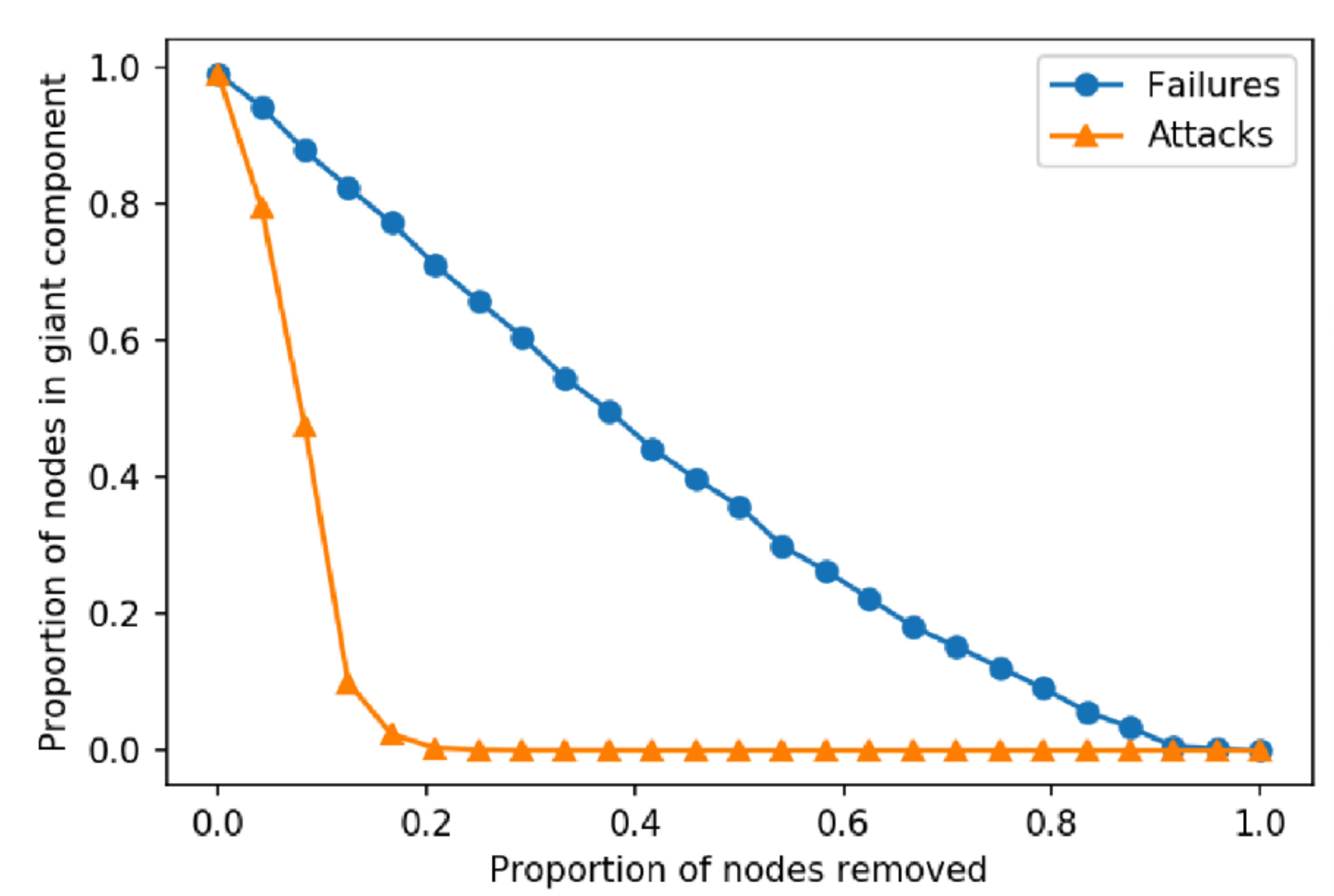
Robustness

- **Two strategies:**
  1. **Random failures:** nodes break down randomly, so they are all chosen with the **same probability**
  2. **Attacks:** hubs are deliberately targeted — the larger the **degree**, the higher the probability of removing the node

- In the first approach, we remove a fraction *f* of nodes, chosen at random

- In the second approach, we remove the fraction *f* of nodes with largest degree, from the one with largest degree downwards

Robustness



Conclusion: real networks are robust against random failures but fragile against targeted attacks!

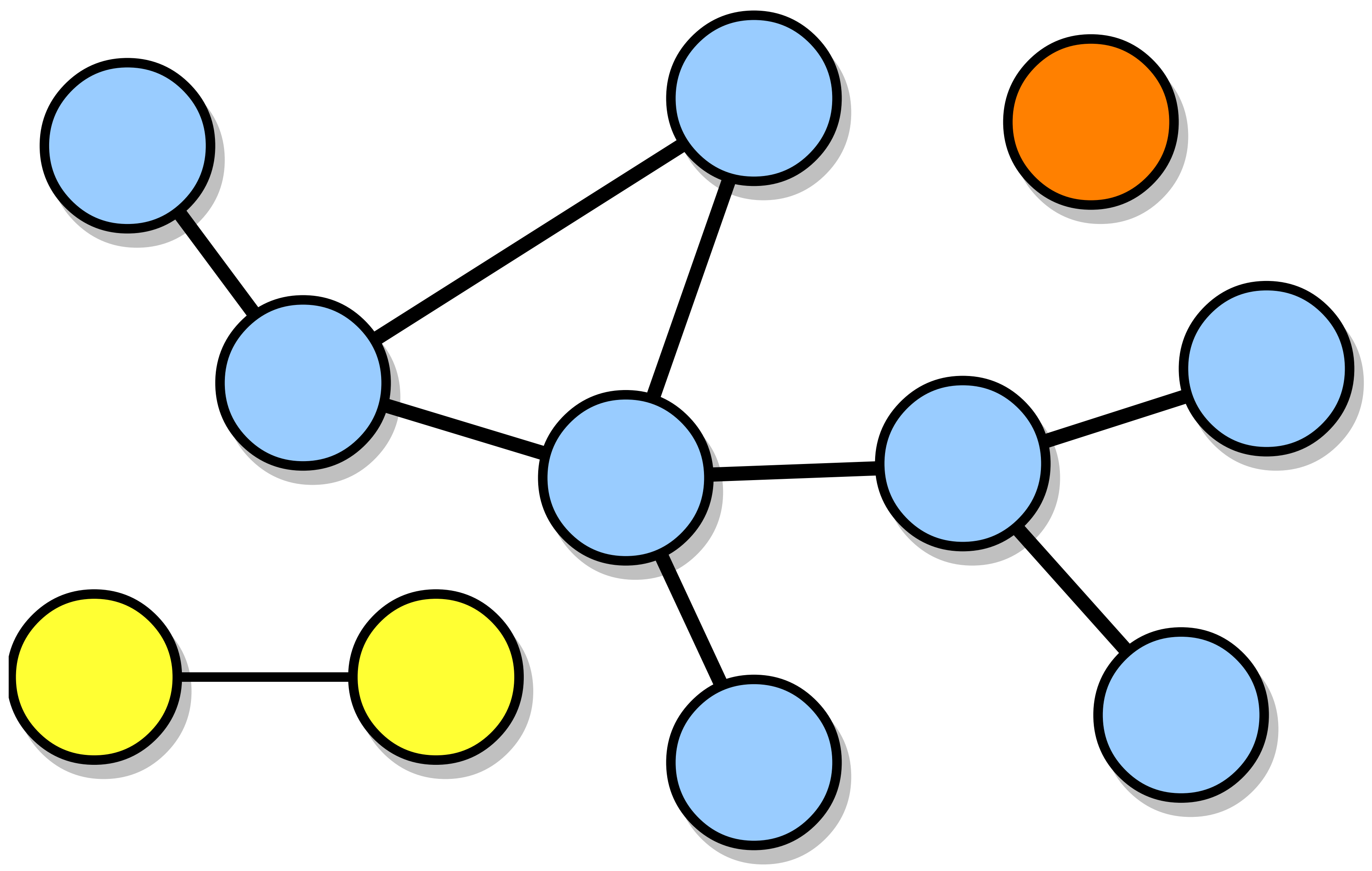
A First Course in Network Science by F. Menczer, S. Fortunato & C.A. Davis. Cambridge University Press, 2020

Pointer to epidemic modeling

- Studying network robustness is a great framework for comparing different immunization (vaccination) strategies
  - this very simple idea has been applied also for mitigating the diffusion of computer viruses
- Problem: real contact network is not usually available…

Connectedness and components

- A network is **connected** if there is a path between any two nodes
- If a network is not connected, it is **disconnected** and has multiple connected components
- A **connected component** is a connected subnetwork
  - The largest one is called **giant component**; it often includes a substantial portion of the network
  - A **singleton** is the smallest-possible connected component



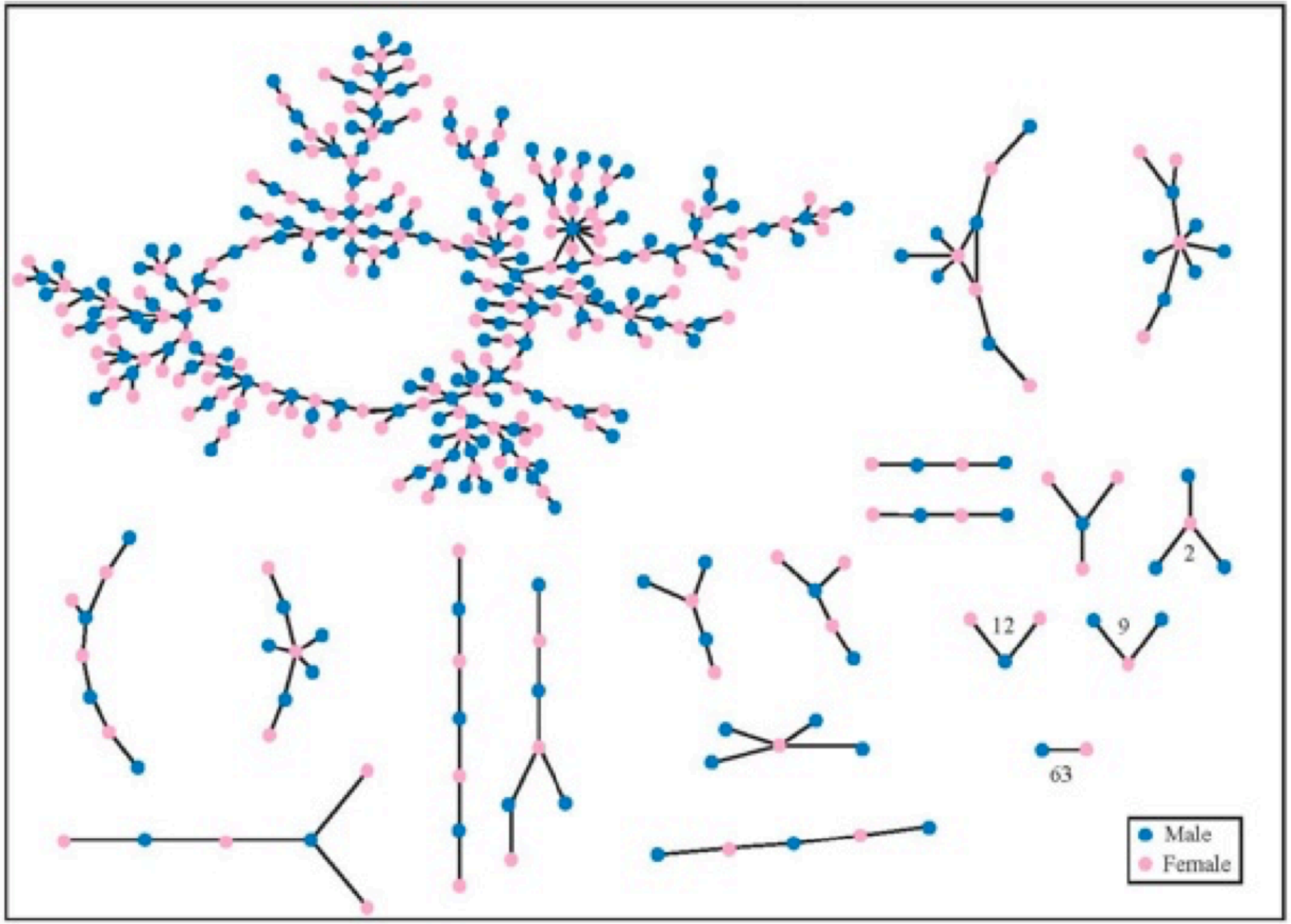
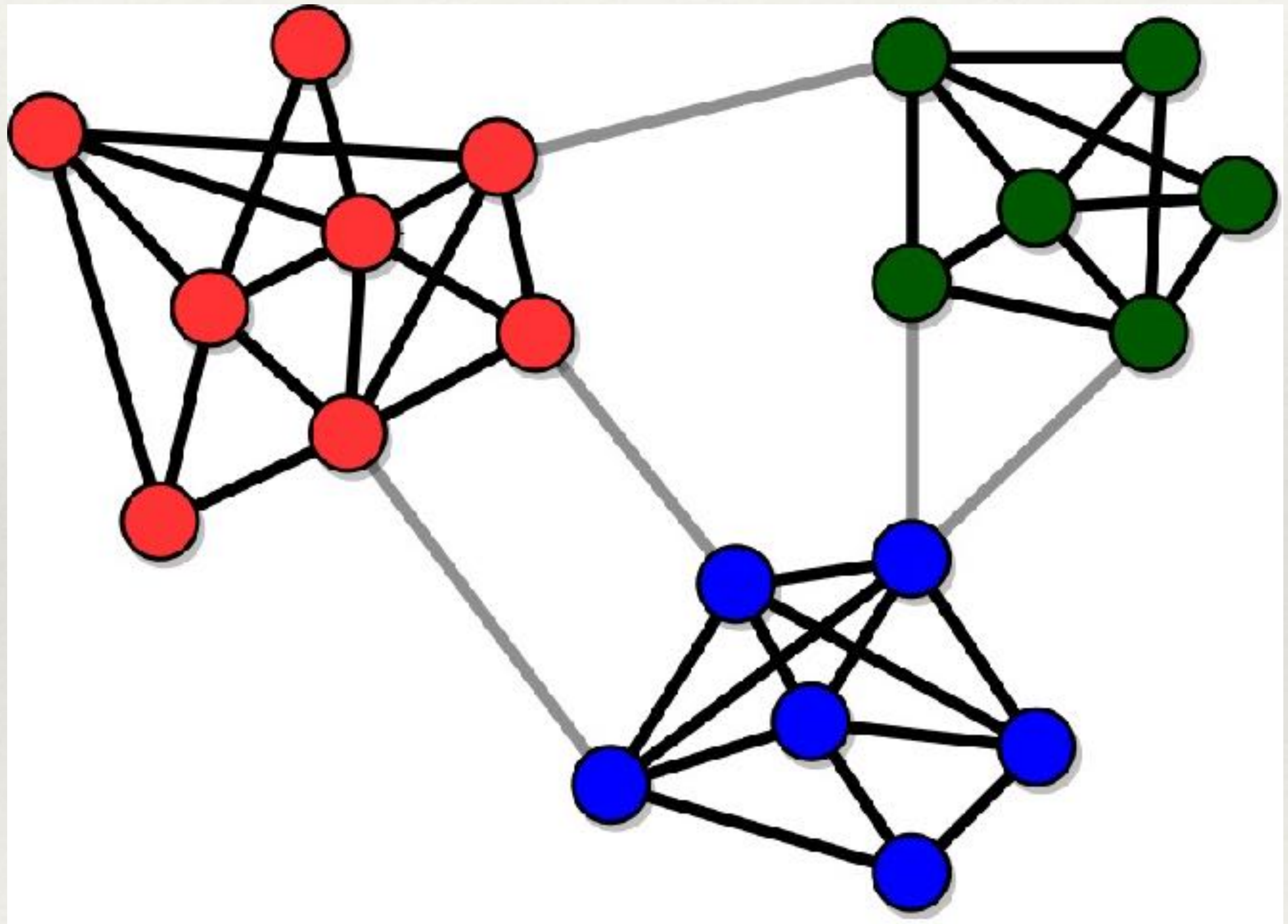


Figure 2.7: A network in which the nodes are students in a large American high school, and an edge joins two who had a romantic relationship at some point during the 18-month period in which the study was conducted [49].

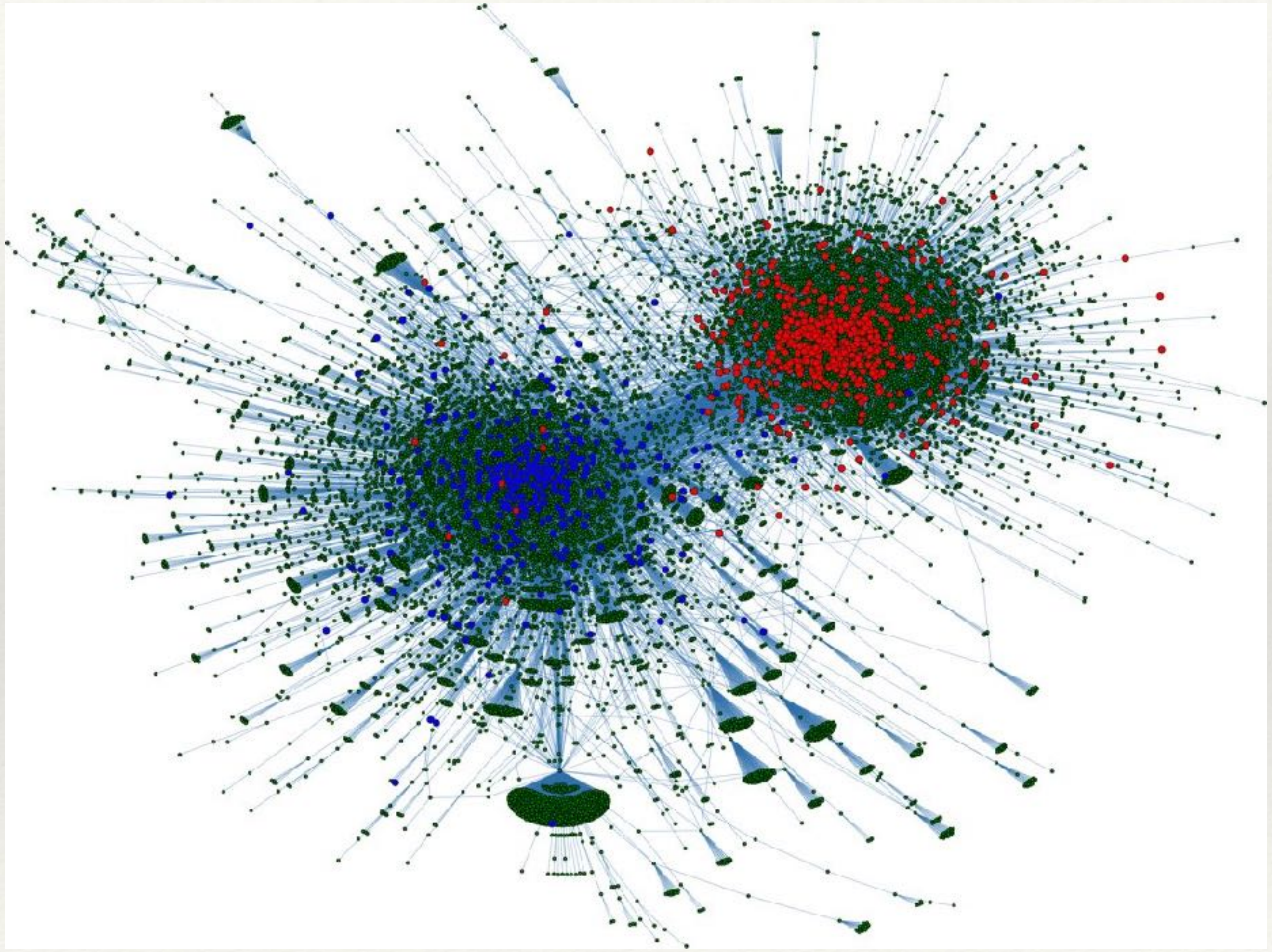
Community structure

Communities (or clusters): sets of tightly connected nodes



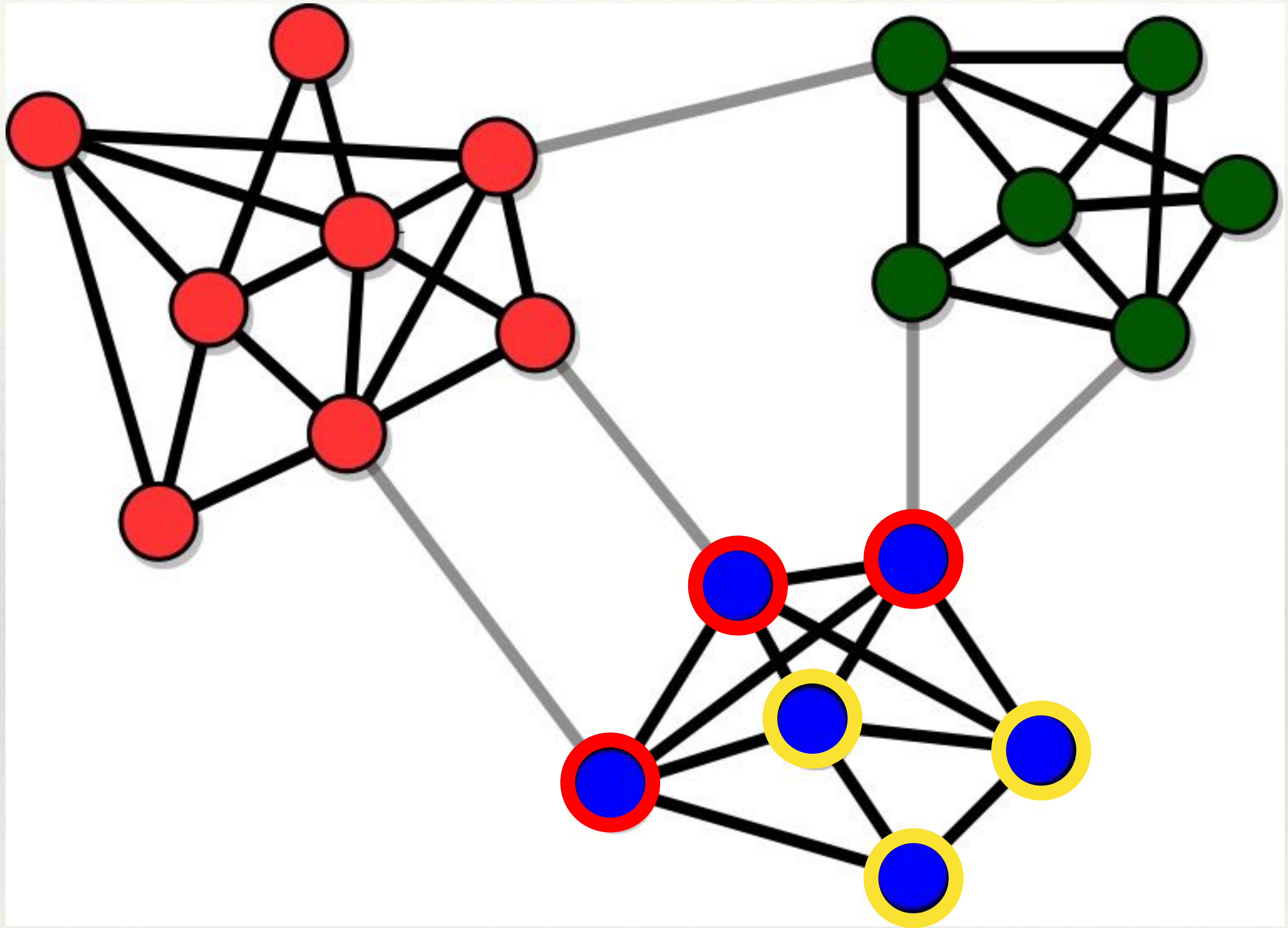
Community structure

- **Example:** Twitter users with strong political preferences tend to follow those aligned with them and not to follow users with different political orientation
- **Other examples:** social circles in social networks, functional modules in protein interaction networks, groups of pages about the same topic on the Web, etc.



Why study communities?

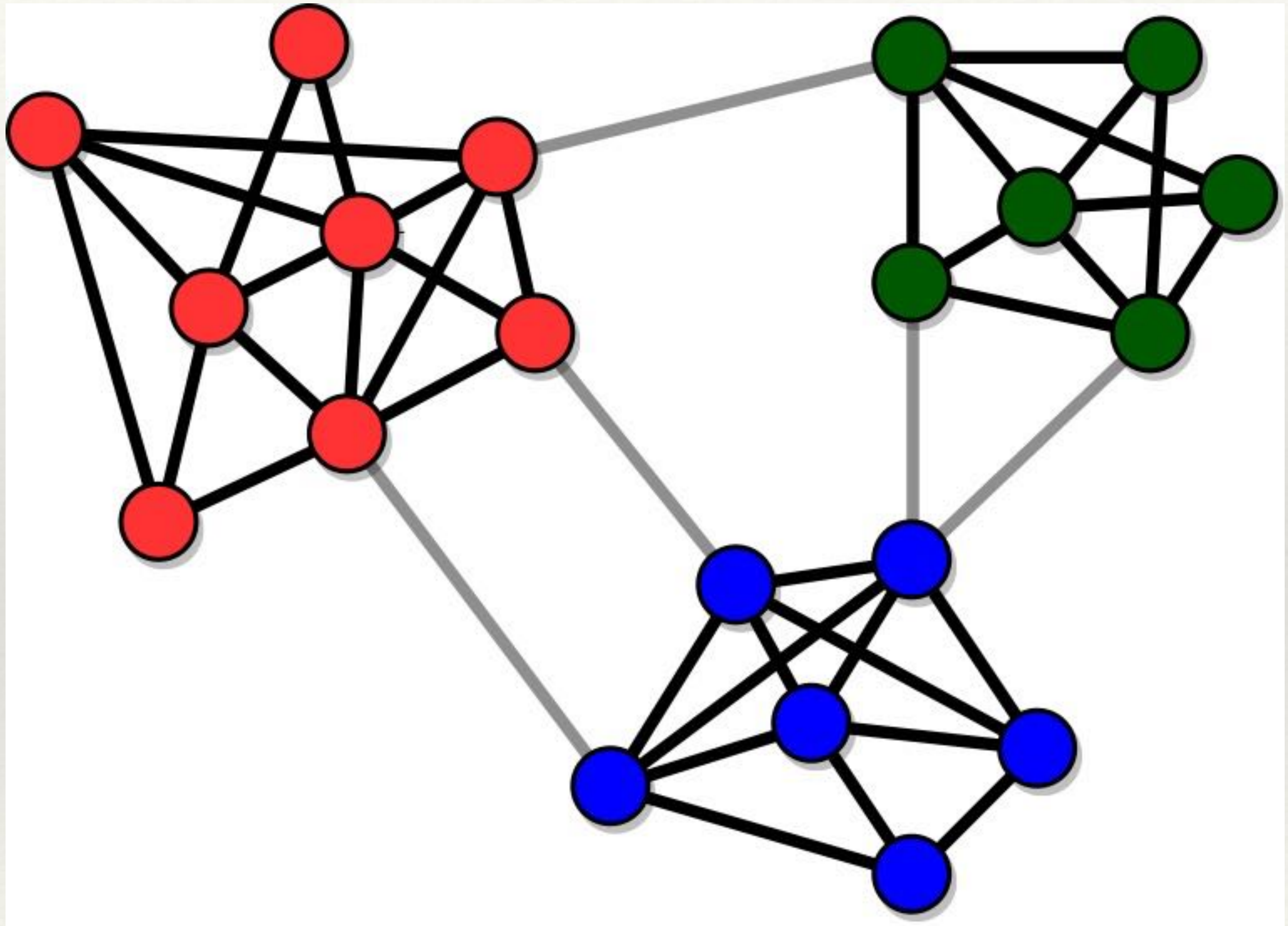
- Uncover the organization of the network
- Identify features of the nodes
- Classify the nodes based on their position in the clusters
- Find missing links



Basic definitions: community

Two main features:

- **High cohesion:** communities have many internal links, so their nodes stick together
- **High separation:** communities are connected to each other by few links



A First Course in Network Science by F. Menczer, S. Fortunato & C.A. Davis. Cambridge University Press, 2020

Partitions

- The number of partitions of n objects is the **Bell number** $\boldsymbol{B_n}$

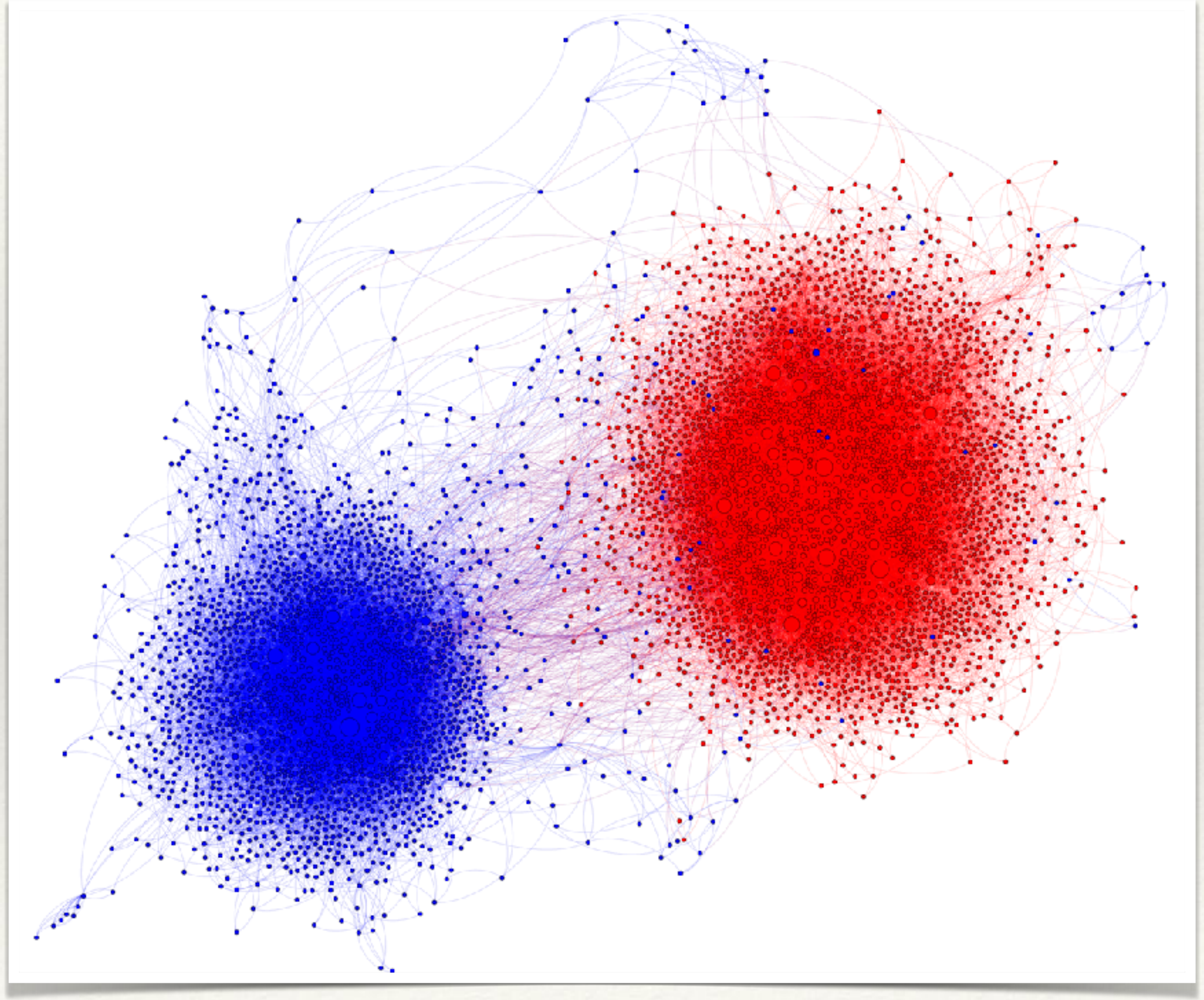
- The Bell number **grows faster than exponentially with** $\boldsymbol{n}$

- **Conclusion:** it makes no sense to look for interesting community structures by exploring the whole space of partitions! A smart exploration of the partition space must be performed.

n	B_n
1	1
2	2
3	5
4	15
5	52
6	203
7	877
8	4140
9	21147
10	115975
11	678570
12	4213597
13	27644437
14	190899322
15	1382958545

example: retweet networks

- goal: detecting communities or clusters
- "**echo chambers**"
- **homophily**: tendency of individuals to link with similar ones
- warning: no trivial linear relationships but **interplay**



Picture from: F. Menczer, S. Fortunato, C. A. Davis, A First Course in Network Science, Cambridge University Press, 2020

Drawing networks

- A **network layout algorithm** places nodes on a plane to visualize the structure of the network
- There are many layout algorithms; the most commonly used are **force-directed layout** (a.k.a. **spring layout**) algorithms:
  - Connected nodes are placed near each other
  - Links have similar length
  - Link crossings are minimized
- This is done by simulating a physical systems where adjacent nodes are connected by springs and otherwise repel each other
- The community structure of the network can be revealed this way if the network is not too dense or too large