Complex Networks
Graph Theory

Vander Luis de Souza Freitas
vander.freitas@ufop.edu.br

UFOP

Agenda

- Last class we briefly discussed:
  - Complex networks
  - Network Models
  - Degree correlations
  - Robustness
  - Community detection
  - Spreading Phenomena

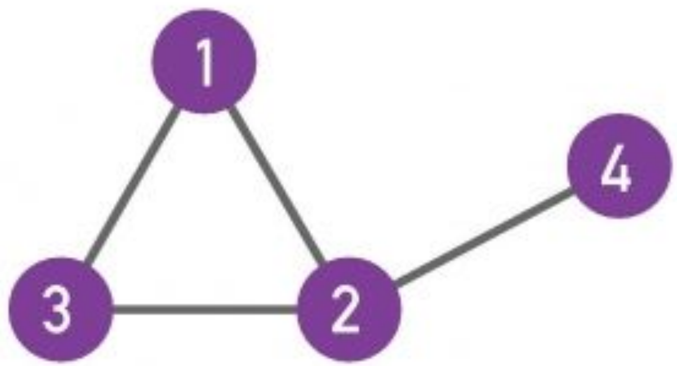
Agenda

- Today:
  - Discussion about the last week homework
  - Graph theory
  - Degree
  - General network characteristics
  - Paths and distances
  - Clustering coefficient
  - Coding!

Discussion

A first program:

1. Choose a programming language and an appropriate library and load the following network (suggestion: Python + igraph or Python + networkx):



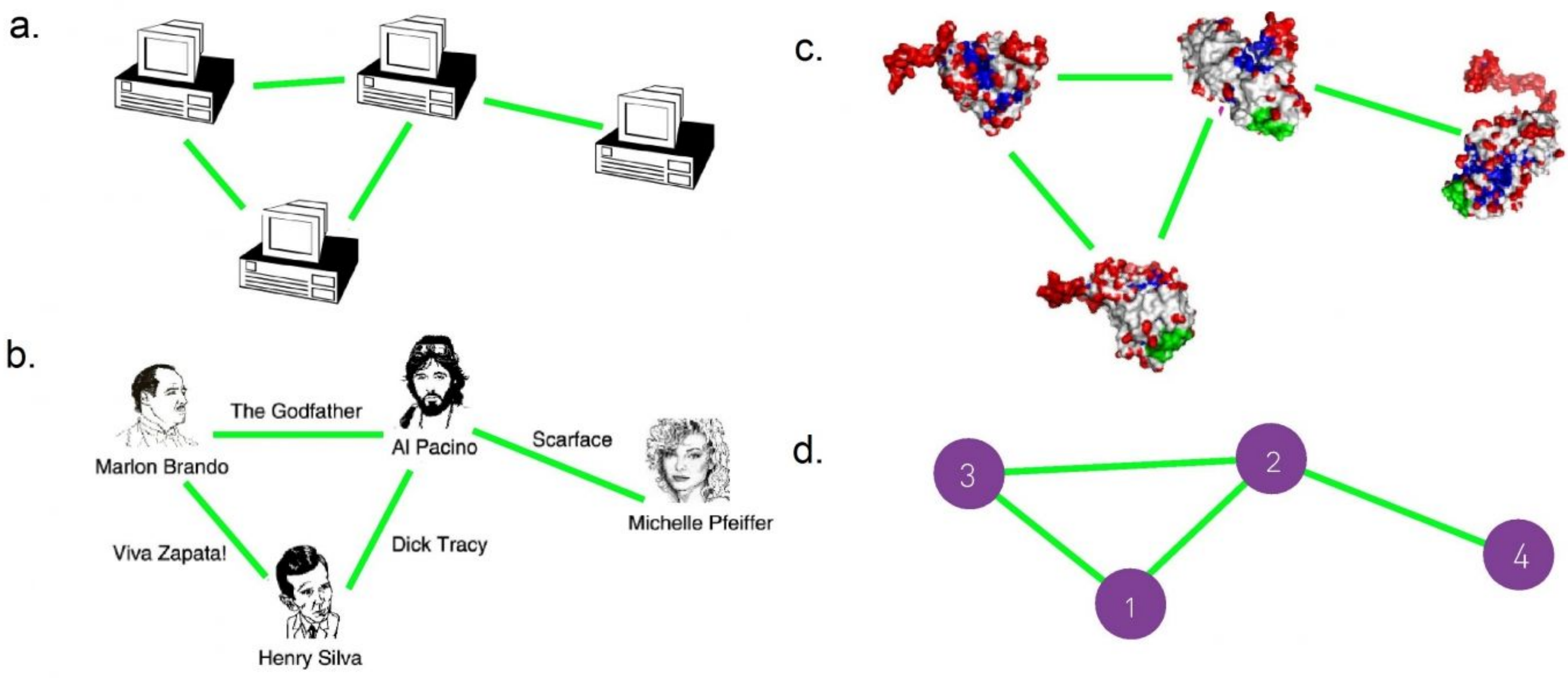
2. Compute its corresponding N and L
3. Plot the network

Graph theory

A graph is represented by $G(V,E)$, where

- V is the set of vertices (nodes)
- E is the set of edges (links)
- $N=|V|$ is the total number of nodes
- $L=|E|$ is the total number of edges

Graph theory



Properties:

- N = 4
- L = 4

http://networksciencebook.com/

Degree

The degree k_i of a node is the number of connections it has.

Undirected networks:

$$L = \frac{1}{2}\sum_{i=1}^{N} k_i$$

The 1/2 factor corrects the problem of counting every link twice.

Average degree:

$$\langle k \rangle = \frac{1}{N}\sum_{i=1}^{N} k_i = \frac{2L}{N}$$

http://networksciencebook.com/

Degree in digraphs

In directed networks (digraphs) we distinguish between:

- Incoming degree k_i^{in} (# of nodes that point to *i*).
- Outgoing degree k_i^{out} (# of nodes that point from *i* to others).

The node's total degree:

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

Total number of links in a directed network:

$$L = \sum_{i=1}^{N} k_i^{in} = \sum_{i=1}^{N} k_i^{out}$$

The average degree of a directed network:

$$\langle k^{in} \rangle = \frac{1}{N} \sum_i^N k^{in} = \langle k^{out} \rangle = \frac{1}{N} \sum_i^N k^{out} = \frac{L}{N}$$

http://networksciencebook.com/

Degree distribution

p_k : probability that a randomly selected node in the network has degree k

As it is a probability:

$$\sum_{k=1}^{\infty} p_k = 1$$

For a network with *N* nodes the degree distribution is the normalized histogram, where

$$p_k = \frac{N_k}{N}$$

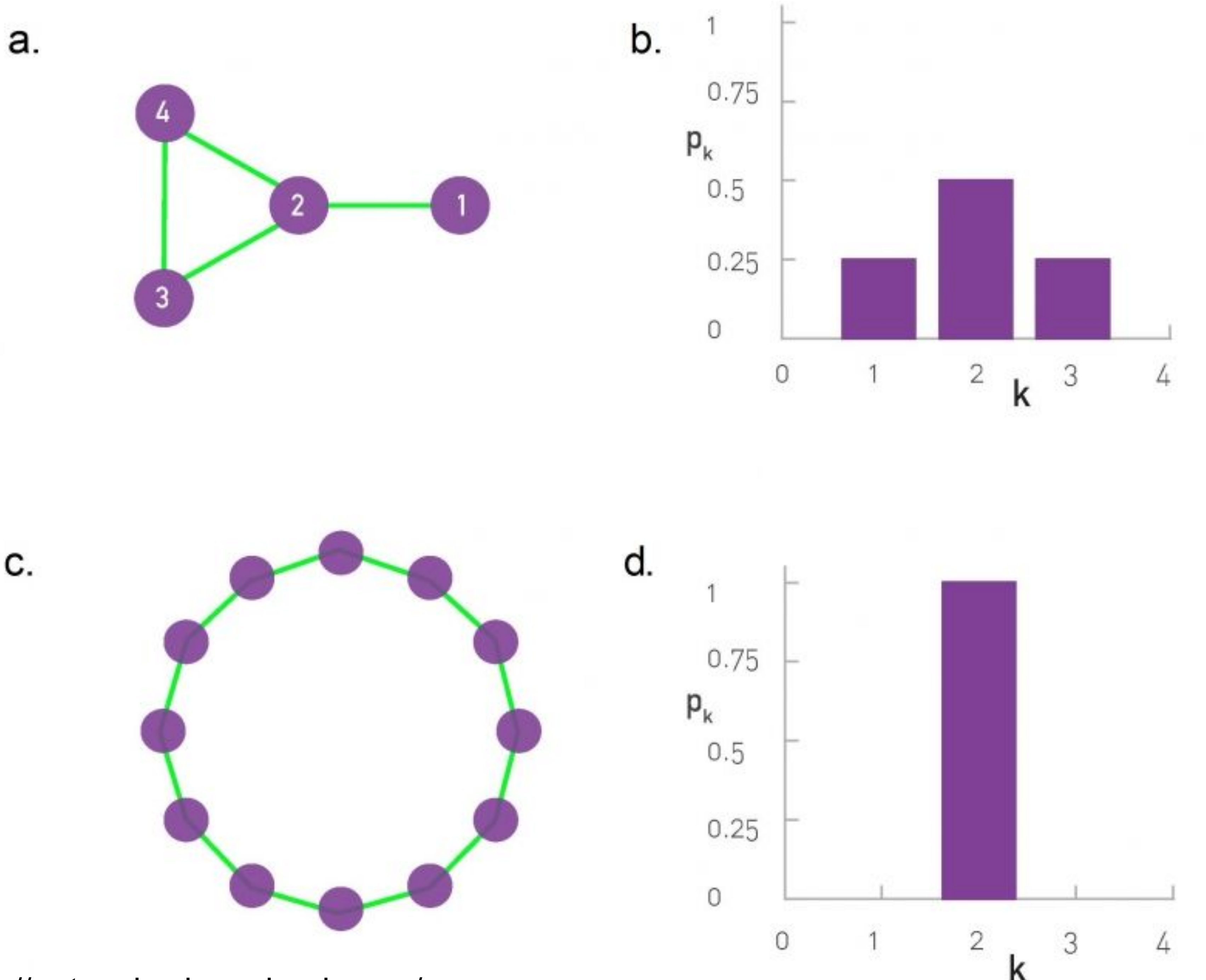
in which N_k is the number of degree-k nodes. Hence $N_k = Np_k$

Average degree from distribution:

$$\langle k \rangle = \sum_{k=0}^{\infty} kp_k$$

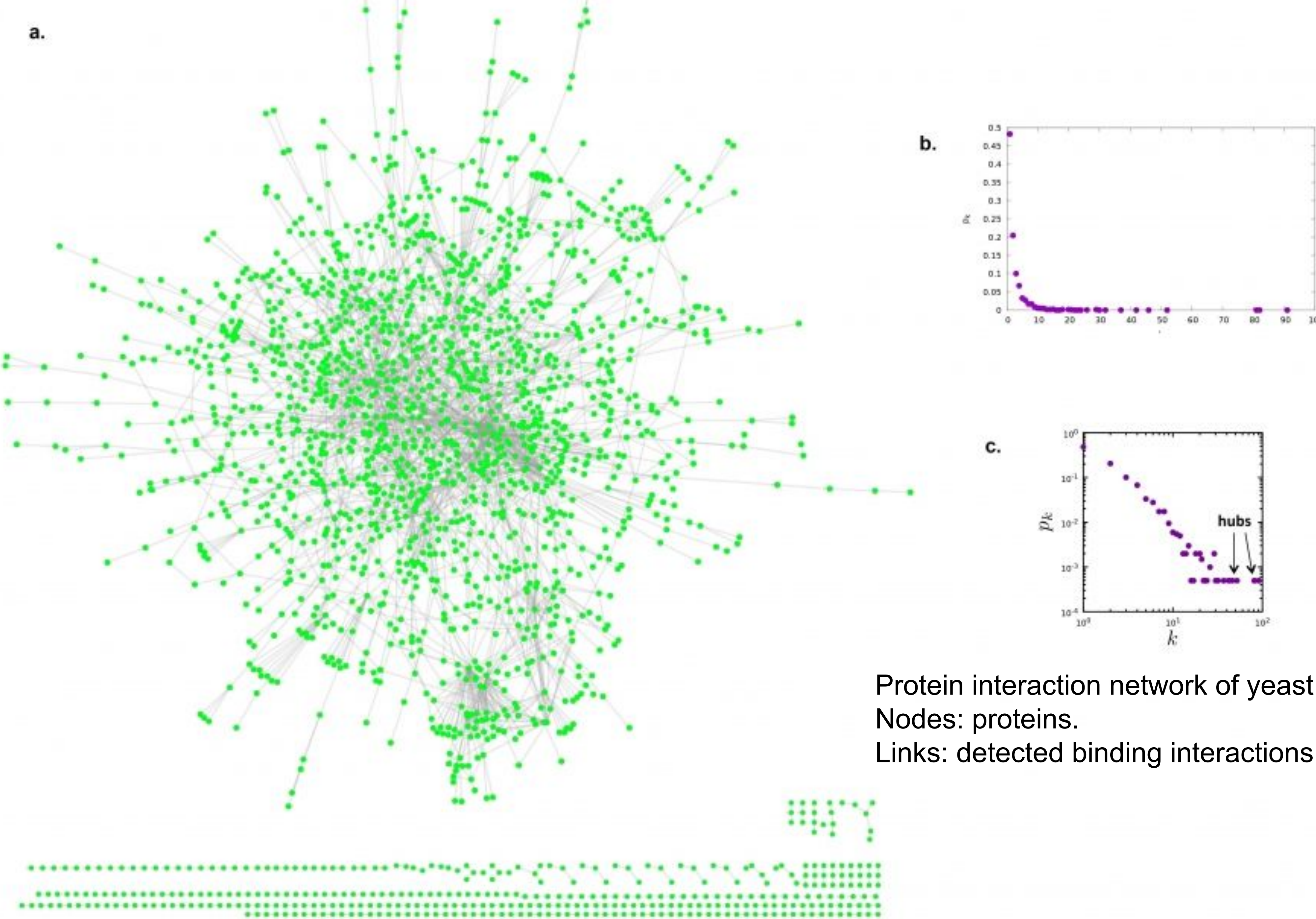
http://networksciencebook.com/

Degree distribution



http://networksciencebook.com/

Degree distribution

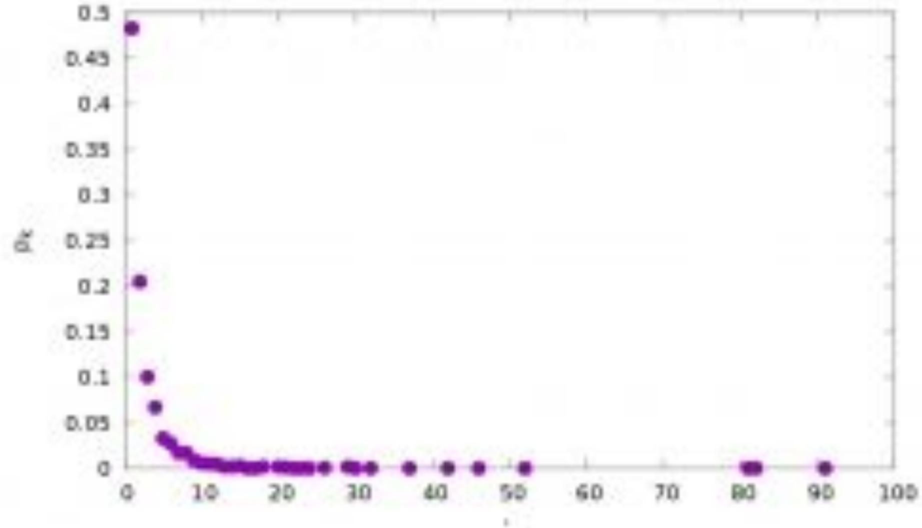


Protein interaction network of yeast.
Nodes: proteins.
Links: detected binding interactions.

http://networksciencebook.com/

Degree distribution

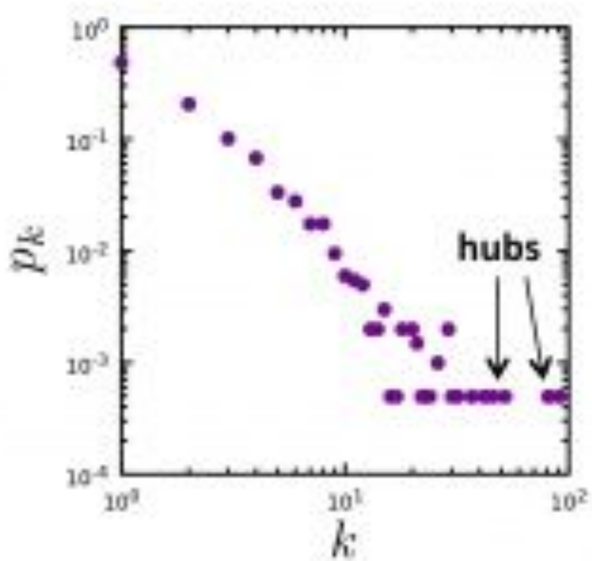
b.



$$p_1 = 0.48$$

$$p_{92} = 0.0005$$

c.



Almost half of the nodes have degree 1 and there is a link with degree 92.

http://networksciencebook.com/

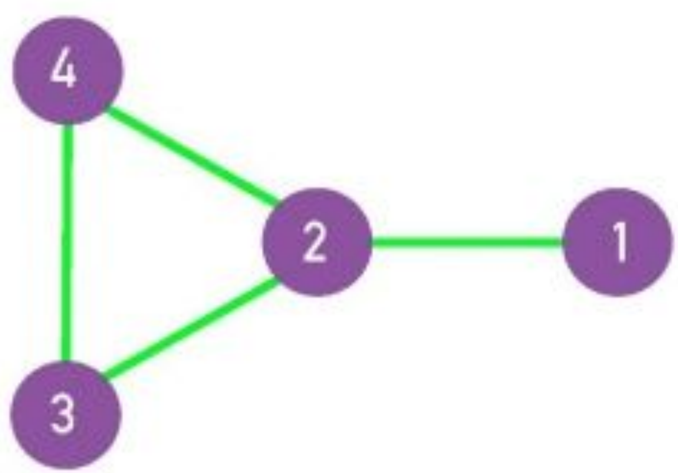
Adjacency matrix

N x N matrix that encodes all network connections, where

$A_{ij} = 1$ if there is a link pointing from node *j* to node *i*.
$A_{ij} = 0$ if nodes *i* and *j* are not connected to each other.

A is symmetric for undirected networks: $A_{ij} = A_{ji}$

Example:



$$A = \begin{pmatrix} 0100 \\ 1011 \\ 0101 \\ 0110 \end{pmatrix}$$

http://networksciencebook.com/

Degree from an Adjacency matrix

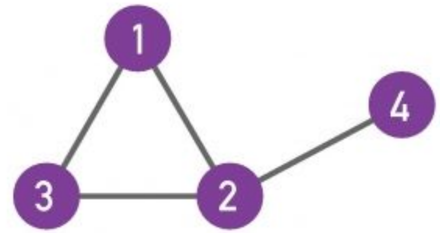
Degree of undirected networks:

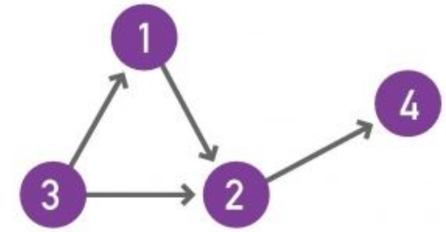
$$k_i = \sum_{j=1}^{N} A_{ij} = \sum_{j=1}^{N} A_{ji}$$

Degree of digraphs:

$$k_i^{in} = \sum_{j=1}^{N} A_{ij} \qquad k_i^{out} = \sum_{j=1}^{N} A_{ji}$$

Degree from an Adjacency matrix





$$A_{ij} = \begin{matrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{matrix}$$

$$A_{ij} = \begin{matrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{matrix}$$

$$k_2 = \sum_{j=1}^{4} A_{2j} = \sum_{i=1}^{4} A_{i2} = 3$$

$$k_2^{\text{in}} = \sum_{j=1}^{4} A_{2j} = 2, \; k_2^{\text{out}} = \sum_{i=1}^{4} A_{i2} = 1$$

$$A_{ij} = A_{ji} \qquad A_{ii} = 0$$

$$A_{ij} \neq A_{ji} \qquad A_{ii} = 0$$

$$L = \frac{1}{2} \sum_{i \;\; i=1}^{N} A_{ij}$$

$$L = \sum_{i,j=1}^{N} A_{ij}$$

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

$$\langle k^{\text{in}} \rangle = \langle k^{\text{out}} \rangle = \frac{L}{N}$$

http://networksciencebook.com/

Real networks are sparse

Network	Nodes	Links	Directed / Undirected	N	L	‹K›
Internet	Routers	Internet connections	Undirected	192,244	609,066	6.34
WWW	Webpages	Links	Directed	325,729	1,497,134	4.60
Power Grid	Power plants, transformers	Cables	Undirected	4,941	6,594	2.67
Mobile-Phone Calls	Subscribers	Calls	Directed	36,595	91,826	2.51
Email	Email addresses	Emails	Directed	57,194	103,731	1.81
Science Collaboration	Scientists	Co-authorships	Undirected	23,133	93,437	8.08
Actor Network	Actors	Co-acting	Undirected	702,388	29,397,908	83.71
Citation Network	Papers	Citations	Directed	449,673	4,689,479	10.43
E. Coli Metabolism	Metabolites	Chemical reactions	Directed	1,039	5,802	5.58
Protein Interactions	Proteins	Binding interactions	Undirected	2,018	2,930	2.90

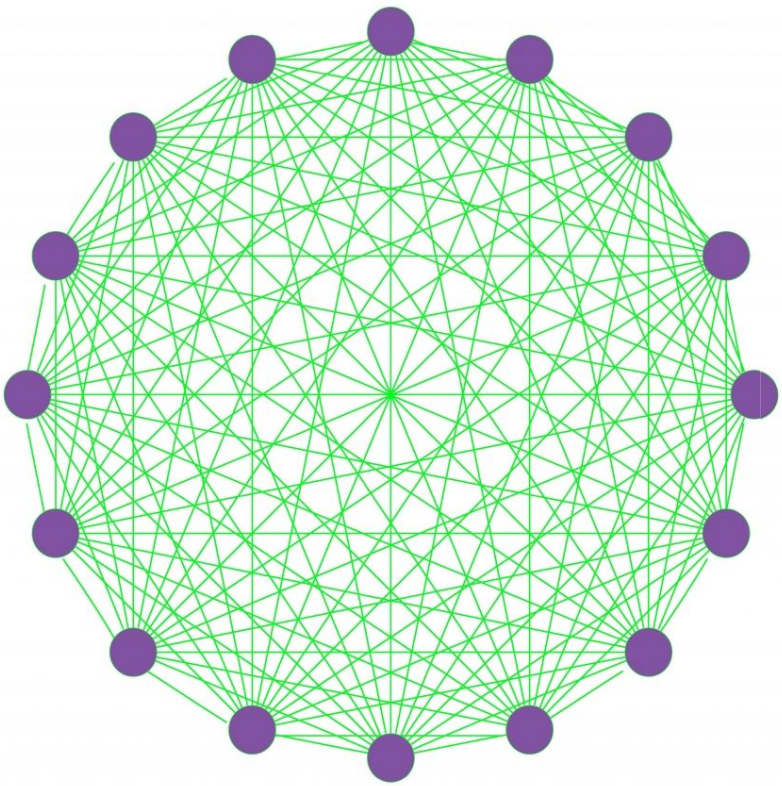
http://networksciencebook.com/

Real networks are sparse

The number of links vary from *L=0* to

$$L_{max} = \frac{N(N-1)}{2}$$

the maximum possible number of links (complete graph).



Example:

$$N = 16, L_{max} = 120$$

$$A_{ij} = 1, \text{for } i, j = \{1, 2, \cdots, N\}$$

$$A_{ii} = 0$$

http://networksciencebook.com/

Real networks are sparse

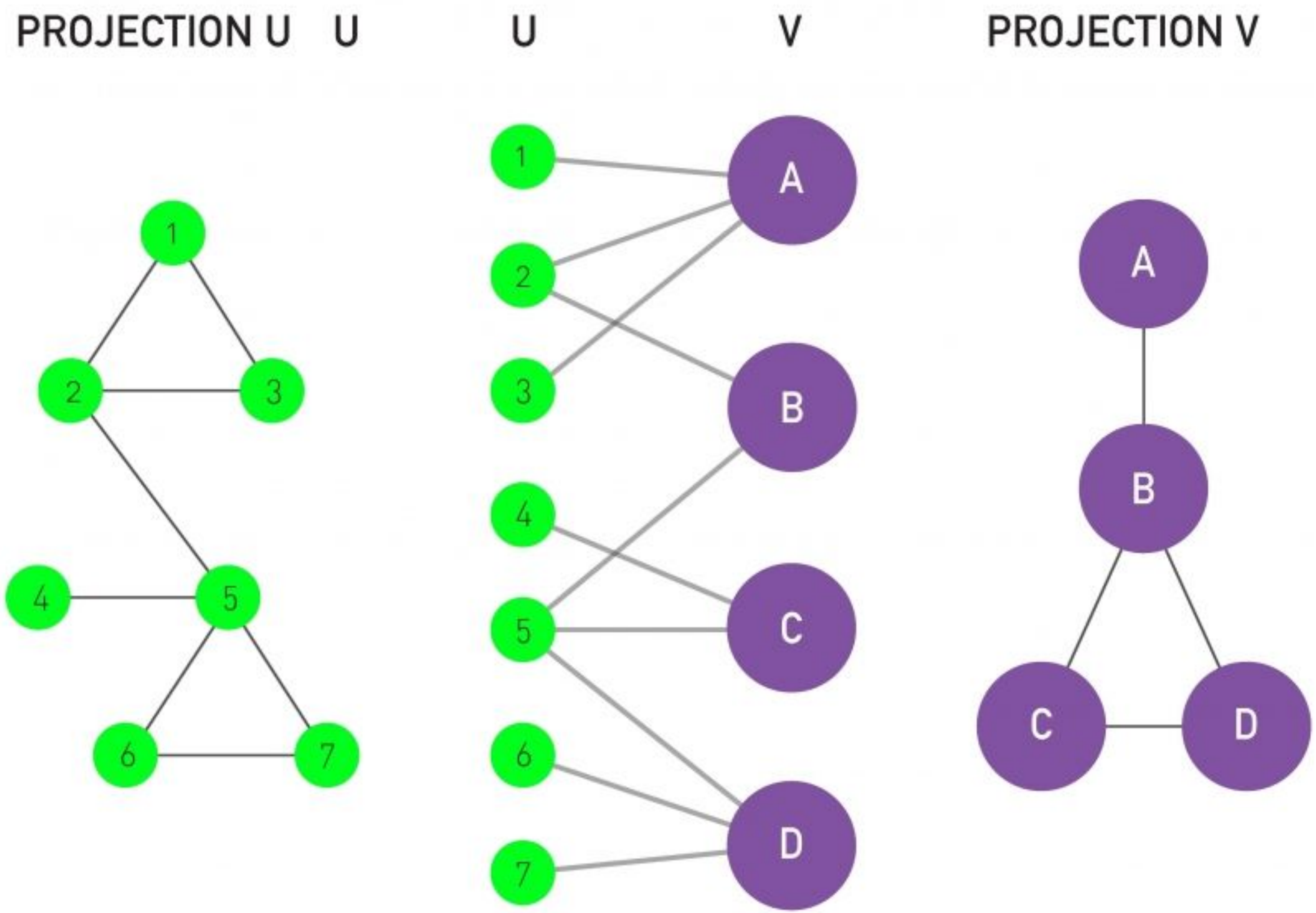
The density of an undirected network is

$$D = \frac{L}{L_{max}} = \frac{2L}{N(N-1)}$$

In a sparse network:

$$L \ll L_{max}\text{, and } D \ll 1.$$

Bipartite networks



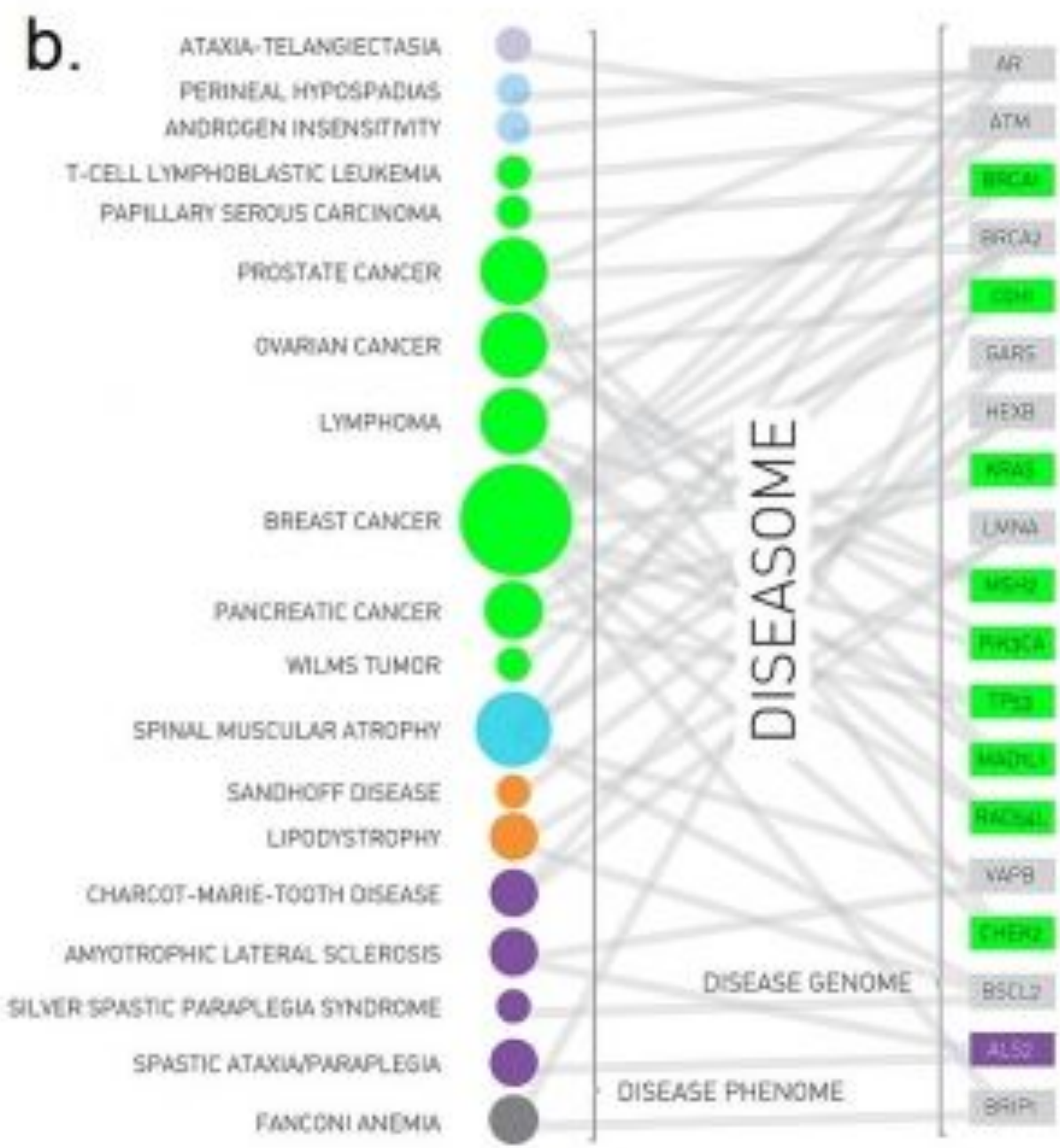
http://networksciencebook.com/

Bipartite networks

Human Disease Network:

A disease is connected to a gene if mutations in that gene are known to affect the particular disease

Diseases

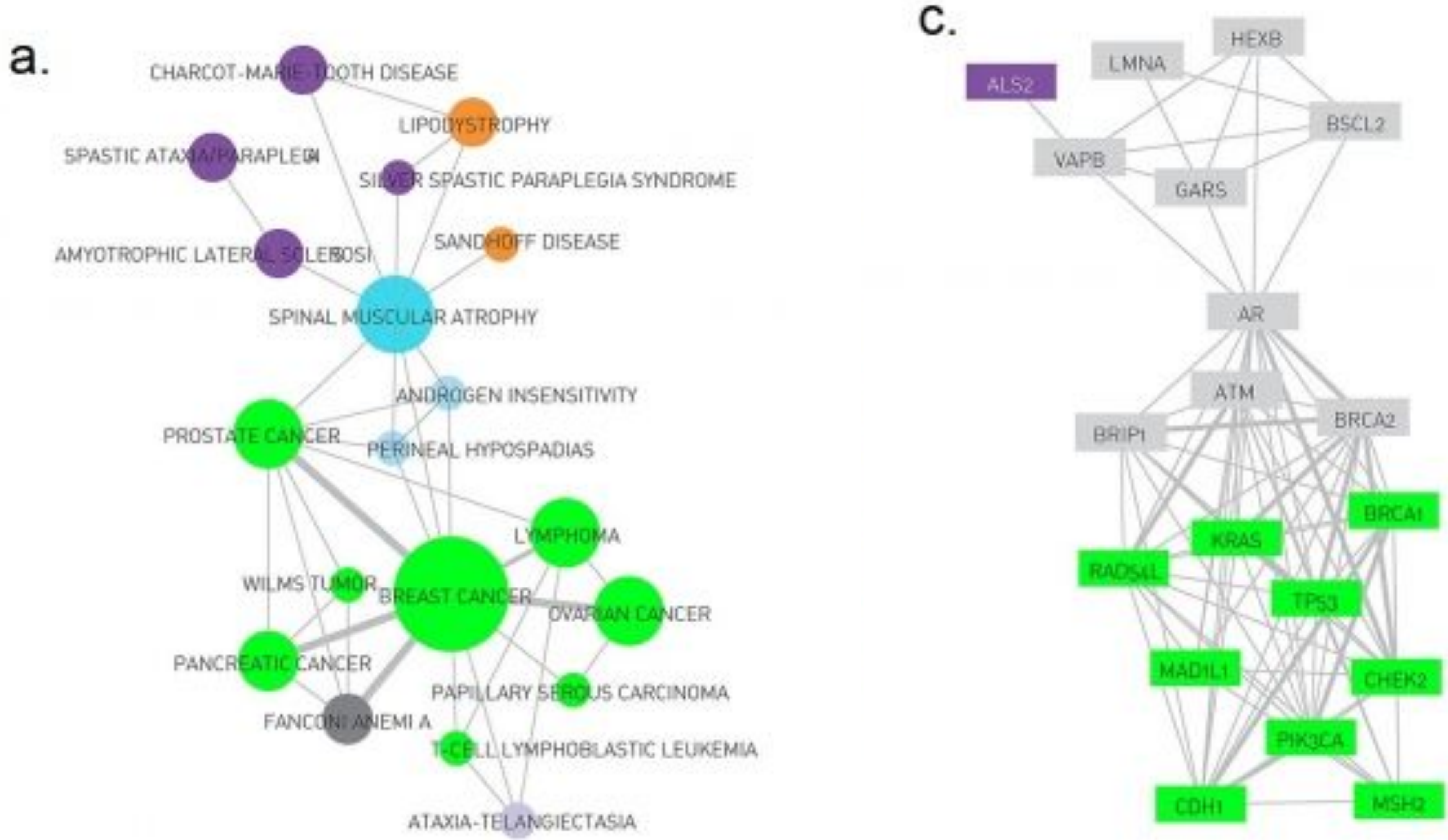


Genes

http://networksciencebook.com/

Bipartite networks

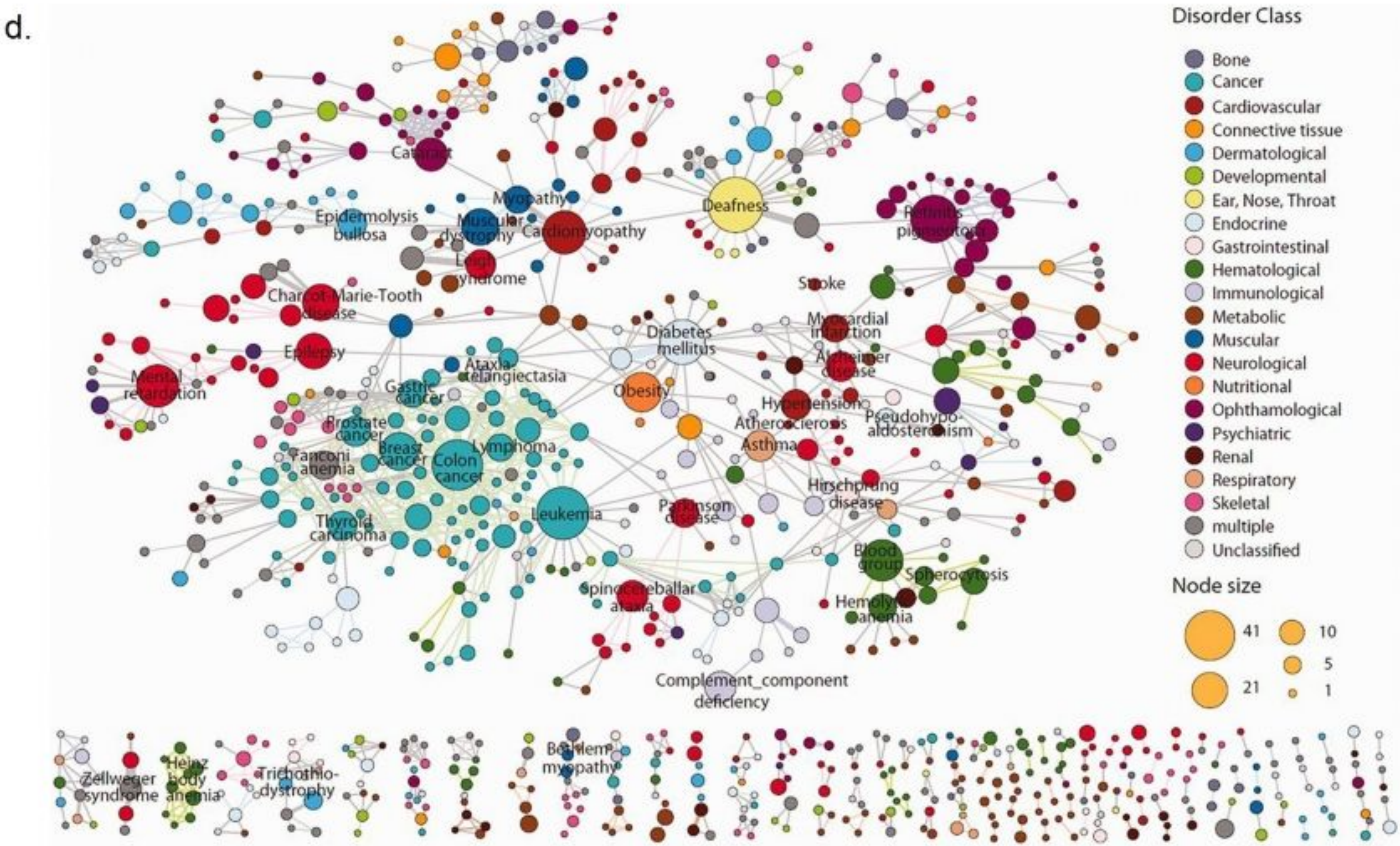
Human Disease Network:



http://networksciencebook.com/

Bipartite networks

Human Disease Network:

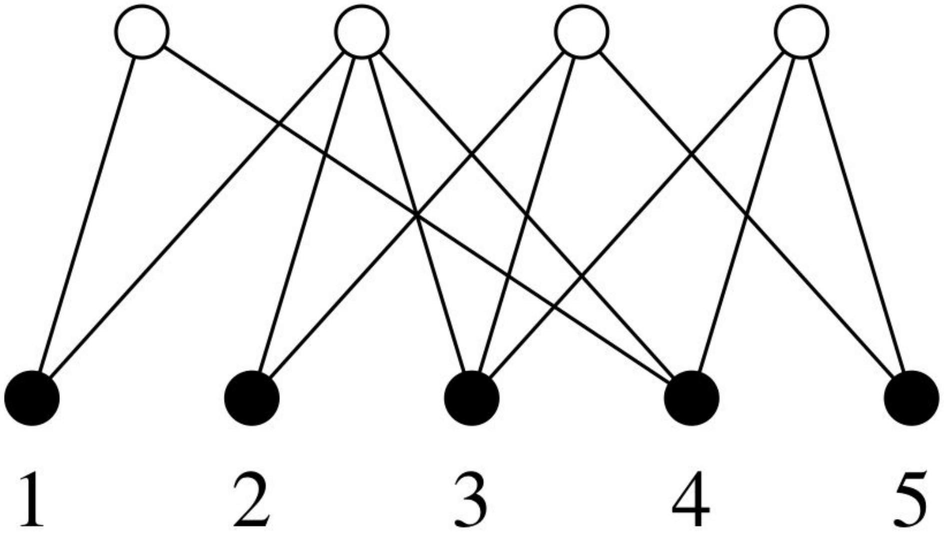


http://networksciencebook.com/

Bipartite networks

Incidence matrix:

$$B_{ij} = \begin{cases} 1 & \text{if item } j \text{ belongs to group } i \\ 0 & \text{otherwise} \end{cases}$$



$$\mathbf{B} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix}$$

(NEWMAN, 2018)

Bipartite networks

The number of groups that nodes *i* and *j* belongs to is:

$$P_{ij} = \sum_{k=1}^{g} B_{ki} B_{kj} = \sum_{k=1}^{g} B_{ik}^{T} B_{kj}$$

The main diagonal of *P* gives the number of groups to which node *i* belongs:

$$P_{ii} = \sum_{k=1}^{g} B_{ki}^{2} = \sum_{k=1}^{g} B_{ki}$$

(NEWMAN, 2018)

Paths and distances

The concept of distance in a network is different from physical distance.

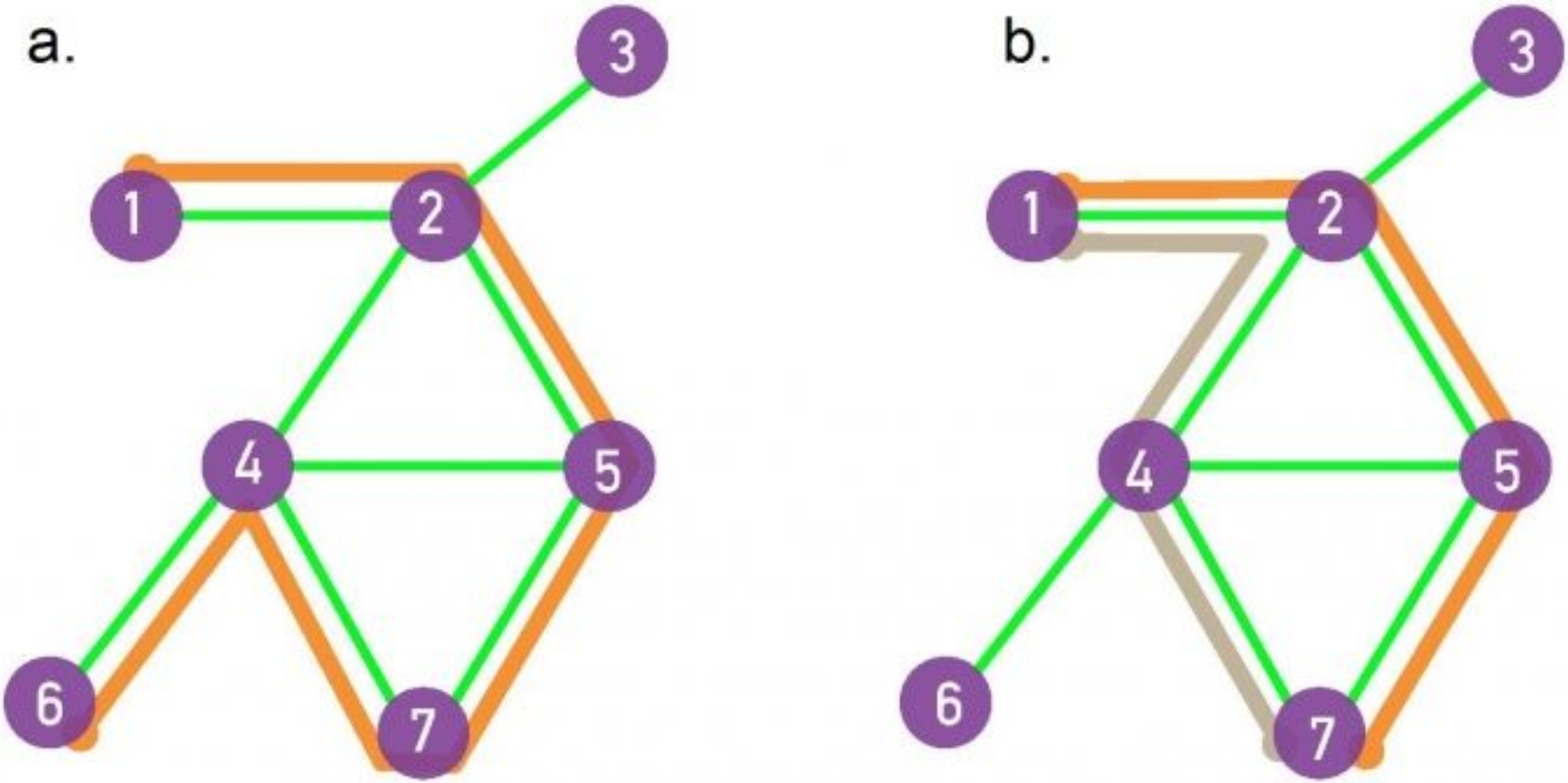
Physical distance:
The distance between two stars has physical implications.

Distance in a network:
Two web pages hosted in servers in opposite sides of the globe may have a direct link in the WWW network. However, two web pages from servers in the same building may not have direct links.

In networks, physical distance is replaced by **Path Length**

Paths and distances

Path: route that runs along the links of the network.
Path length: the number of links the path contains.



http://networksciencebook.com/

Shortest path (or geodesics)

Shortest path between nodes *i* and *j*: the path with the fewest number of links. Usually computed with the Breadth-First Search (BFS) Algorithm.

It is often denoted by d_{ij}

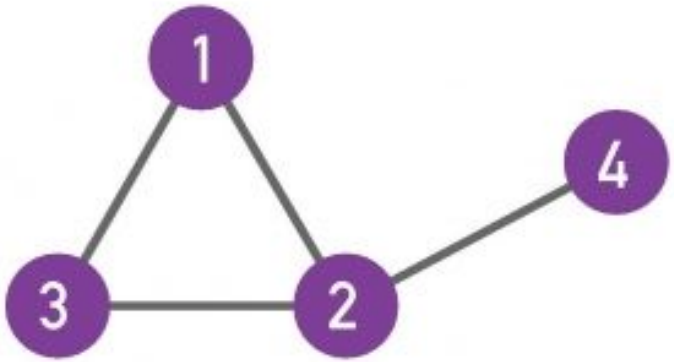
The shortest path never contains loops or intersects itself.

In an undirected network $d_{ij} = d_{ji}$

In a digraph, usually $d_{ij} \neq d_{ji}$

Network diameter: longest shortest path in a graph d_{max}

Path length



$$A^3 = \begin{pmatrix} 2 & 4 & 3 & 1 \\ 4 & 2 & 4 & \boxed{3} \\ 3 & 4 & 2 & 1 \\ 1 & 3 & 1 & 0 \end{pmatrix}$$

$$A^2 = \begin{pmatrix} 2 & 1 & 1 & 1 \\ 1 & 3 & 1 & \boxed{0} \\ 1 & 1 & 2 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix}$$

$$A_{ij} = \begin{matrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & \boxed{1} \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{matrix}$$

Paths of length 1 between 2 and 4: **1**
Paths of length 2 between 2 and 4: **0**
Paths of length 3 between 2 and 4: **3**

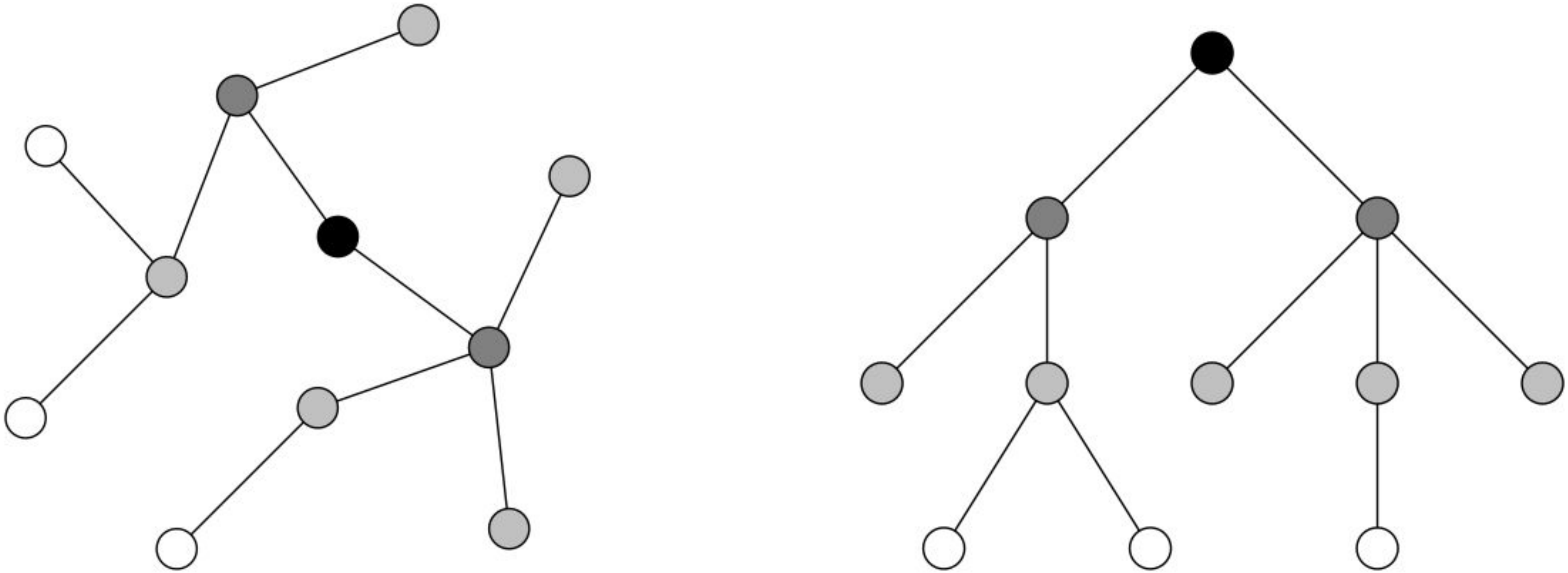
The number of paths of length *d* between *i* and *j* is

$$N_{ij}^{d} = A_{ij}^{d}$$

http://networksciencebook.com/

Trees

Two visualizations of the same tree:

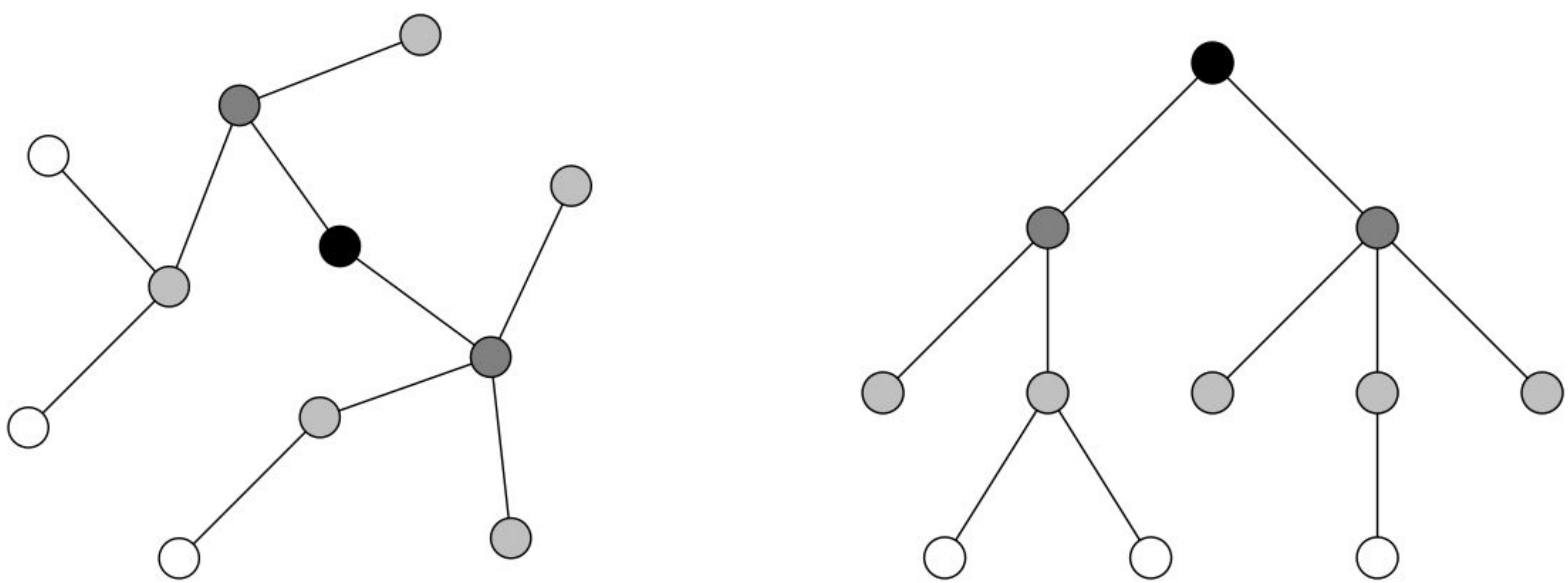


- Trees are connected, undirected, planar, and contains no loops:
  - No loops means that there is only one path between a given pair of nodes.
- Nodes with only one neighbor are called leaves.

(NEWMAN, 2018)

Trees

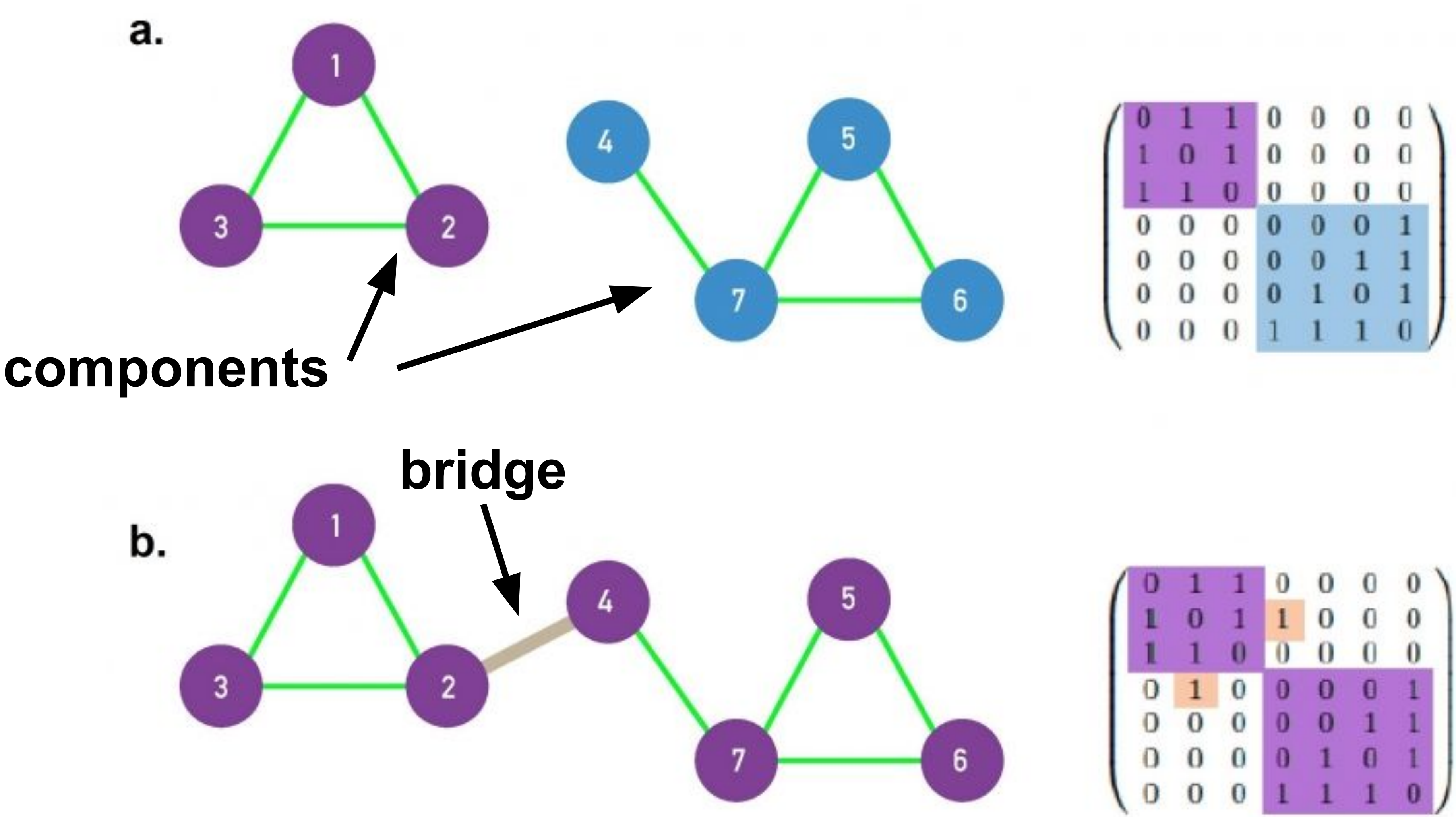
- A tree of N nodes has exactly N-1 edges;
- Every network of N nodes and N-1 edges is a tree;
- This implies that a connected network of N nodes with the minimum number of edges is always a tree;
- Examples of trees: rivers, dendrograms, some data structures, etc.



(NEWMAN, 2018)

Connectedness

A network is connected if there exists a path between any pair of nodes *i* and *j*.
If it does not exist, $d_{ij} = \infty$



http://networksciencebook.com/

Clustering coefficient

Are my friends also friends with each other?

The clustering coefficient captures the degree to which the neighbors of a given node link to each other. The local clustering coefficient of a node *i* is:

$$C_i = \frac{2L_i}{k_i(k_i-1)}$$

in which L_i is the number of links between the k_i neighbors of node *i*. $C_i \in [0,1]$ is the probability that two neighbors of a node link to each other.

http://networksciencebook.com/

Clustering coefficient

Problem: How to compute the clustering coefficient for nodes with degrees 0 or 1?

One could compute the Global Clustering Coefficient for the whole network (it is not a local metric):

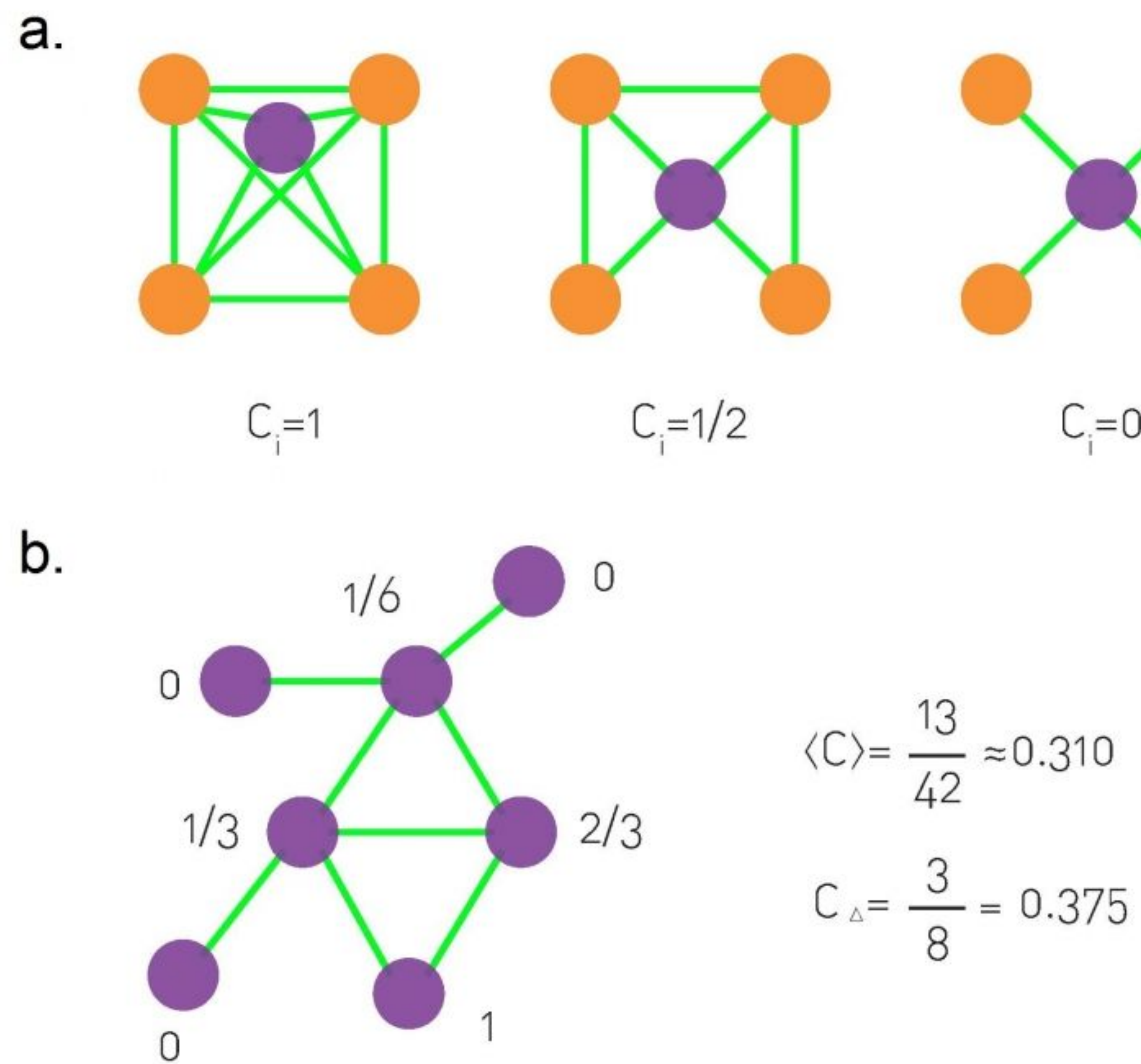
$$C_\Delta = \frac{3 \times \text{NumberOfTriangles}}{\text{NumberOfConnectedTriples}}$$

A connected triplet is an ordered set of three nodes ABC such that A connects to B and B connects to C. For example, an A, B, C triangle is made of three triplets, ABC, BCA and CAB.

C_Δ and $\langle C_i \rangle$ are usually different.

http://networksciencebook.com/

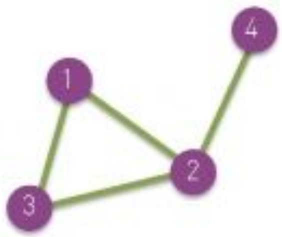
Clustering coefficient



http://networksciencebook.com/

Summary

a. Undirected

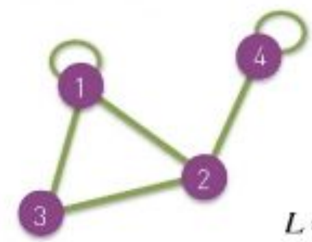


$$A_{ij}=\begin{pmatrix}0&1&1&0\\1&0&1&1\\1&1&0&0\\0&1&0&0\end{pmatrix}$$

$A_{ii}=0 \qquad A_{ij}=A_{ji}$

$L=\frac{1}{2}\sum_{i,j=1}^{N}A_{ij} \qquad <k>=\frac{2L}{N}$

b. Self-loops

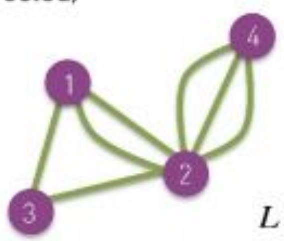


$$A_{ij}=\begin{pmatrix}1&1&1&0\\1&0&1&1\\1&1&0&0\\0&1&0&1\end{pmatrix}$$

$\exists i, A_{ii}\neq 0 \qquad A_{ij}=A_{ji}$

$L=\frac{1}{2}\sum_{i,j=1,i\neq j}^{N}A_{ij}+\sum_{i=1}^{N}A_{ii} \qquad ?$

c. Multigraph
(undirected)

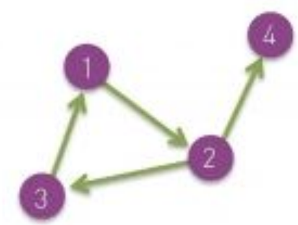


$$A_{ij}=\begin{pmatrix}0&2&1&0\\2&0&1&3\\1&1&0&0\\0&3&0&0\end{pmatrix}$$

$A_{ii}=0 \qquad A_{ij}=A_{ji}$

$L=\frac{1}{2}\sum_{i,j=1}^{N}A_{ij} \qquad <k>=\frac{2L}{N}$

d. Directed

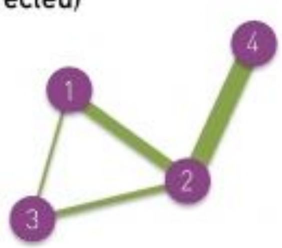


$$A_{ij}=\begin{pmatrix}0&1&0&0\\0&0&1&1\\1&0&0&0\\0&0&0&0\end{pmatrix}$$

$A_{ij}\neq A_{ji}$

$L=\sum_{i,j=1}^{N}A_{ij} \qquad <k>=\frac{L}{N}$

e. Weighted
(undirected)

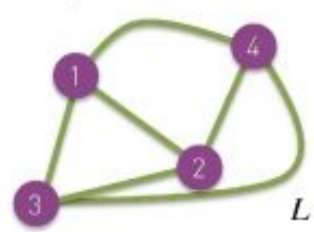


$$A_{ij}=\begin{pmatrix}0&2&0.5&0\\2&0&1&4\\0.5&1&0&0\\0&4&0&0\end{pmatrix}$$

$A_{ii}=0 \qquad A_{ij}=A_{ji}$

$<k>=\frac{2L}{N}$

f. Complete Graph
(undirected)



$$A_{ij}=\begin{pmatrix}0&1&1&1\\1&0&1&1\\1&1&0&1\\1&1&1&0\end{pmatrix}$$

$A_{ii}=0 \qquad A_{i\neq j}=1$

$L=L_{max}=\frac{N(N-1)}{2} \qquad <k>=N-1$

http://networksciencebook.com/

Coding

- Python libraries:
  https://igraph.org/python/doc/tutorial/tutorial.html
  https://networkx.org/documentation/stable/tutorial.html

- Useful code (networkx) from:
  MENCZER, F.; FORTUNATO, S.; DAVIS, C. A First Course in Network Science. Cambridge: Cambridge University Press, 2020. 300 p.

  https://github.com/cambridgeuniversitypress/firstcoursenetworkscience

- Useful code (igraph and networkx) from myself and students:
  https://github.com/vanderfreitas/useful-network-science-code

Coding

Datasets:

- http://networksciencebook.com/translations/en/resources/data.html
- https://networkrepository.com/
- https://snap.stanford.edu/data/
- https://kateto.net/2016/05/network-datasets/