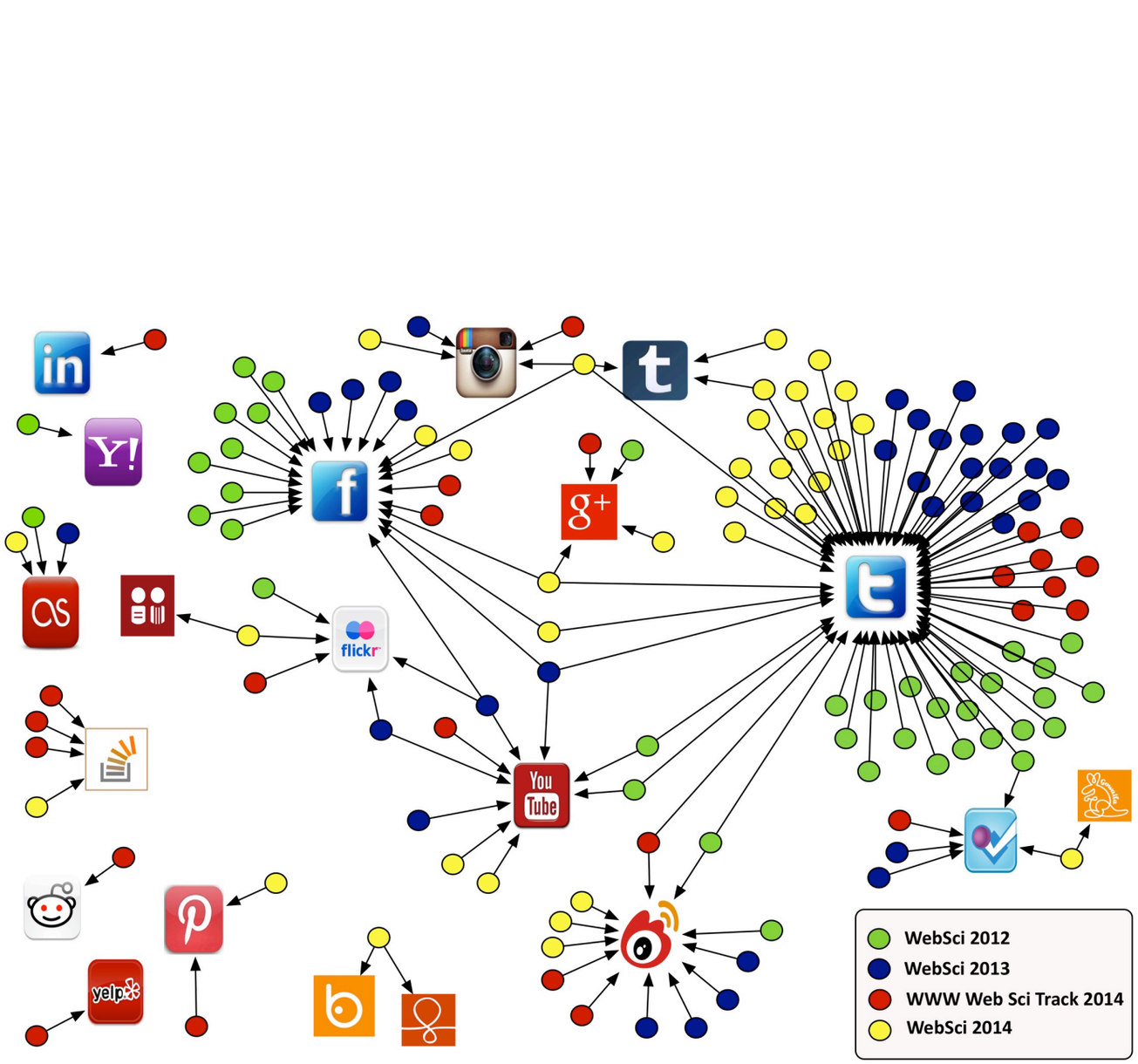
Machine Learning and Modeling for Social Networks

Olivia Woolley Meza, Izabela Moise, Nino Antulov-Fatulin, Lloyd Sanders

Introduction to Networks

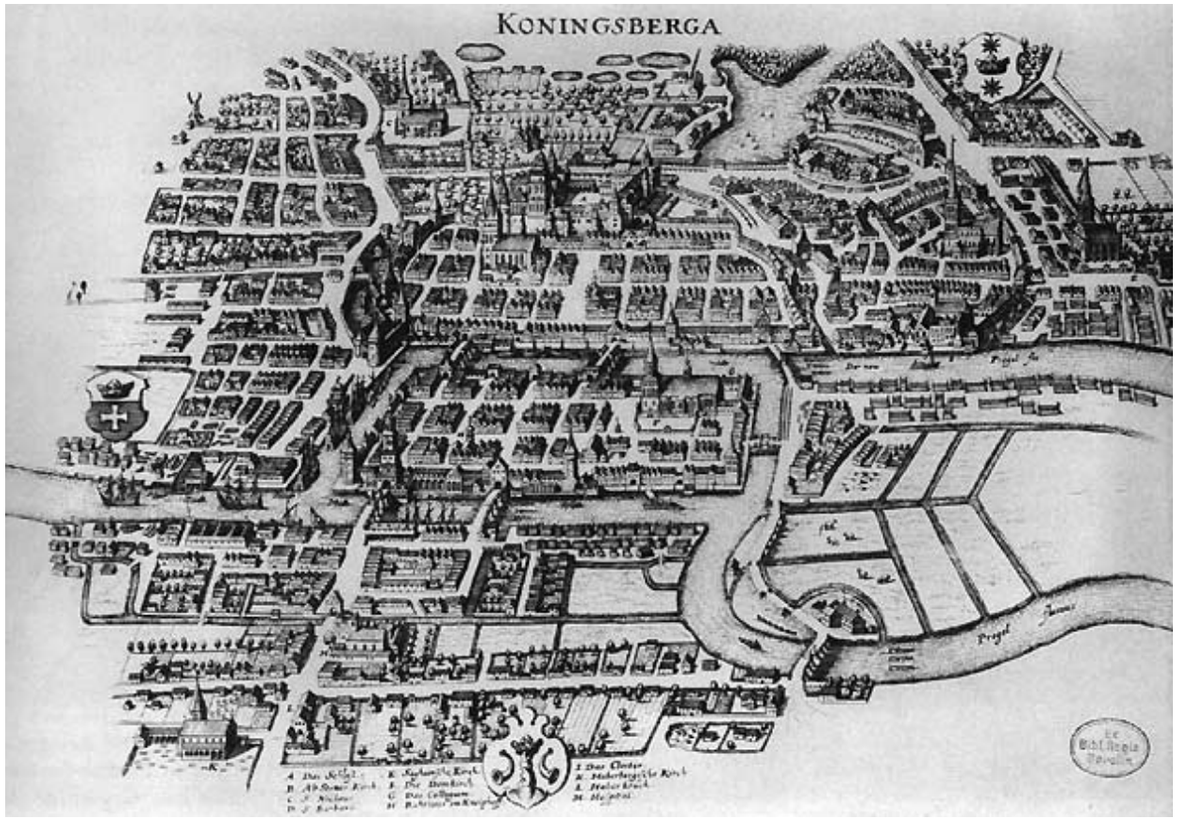
- Motivation
- Basic concepts and definitions
    - Adjacency matrix, paths, connected components
- Centrality
    - Degree, closeness, Page Rank, betweenness
- Structural features (of social networks)
    - Heterogeneity, assortativity, clustering, small world, communities
- Network models
    - Random graphs, generative models

Multiple interconnected social media platforms



source: H. Alani, M. Rowe, Mining and Comparing Engagement Dynamics Across Multiple Social Media Platforms, ACM Web Science Conference (WebSci) 2014.

Seven bridges of Königsberg



In 1736 Euler posted the following problem: *Is it possible to have a walk in the city of Königsberg, that crosses each of the seven bridges only once?*

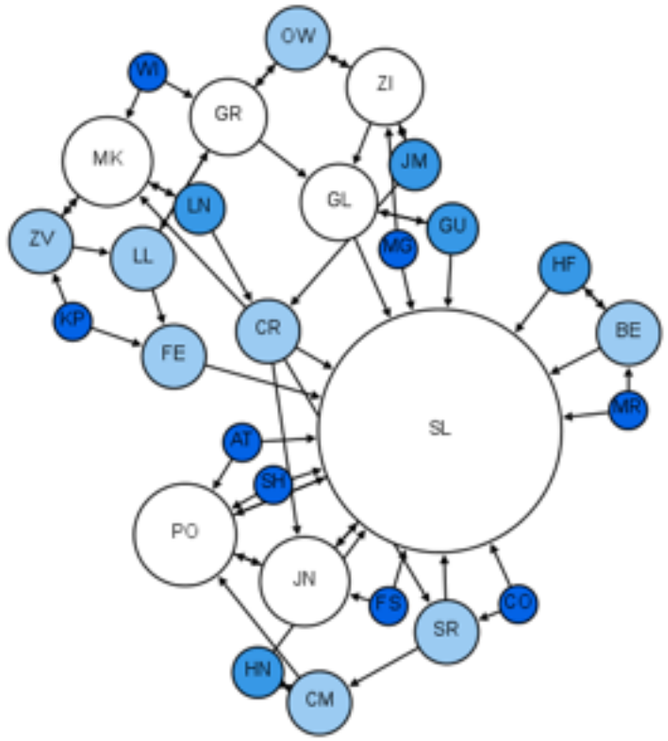
Networks: abstraction and representation of relations

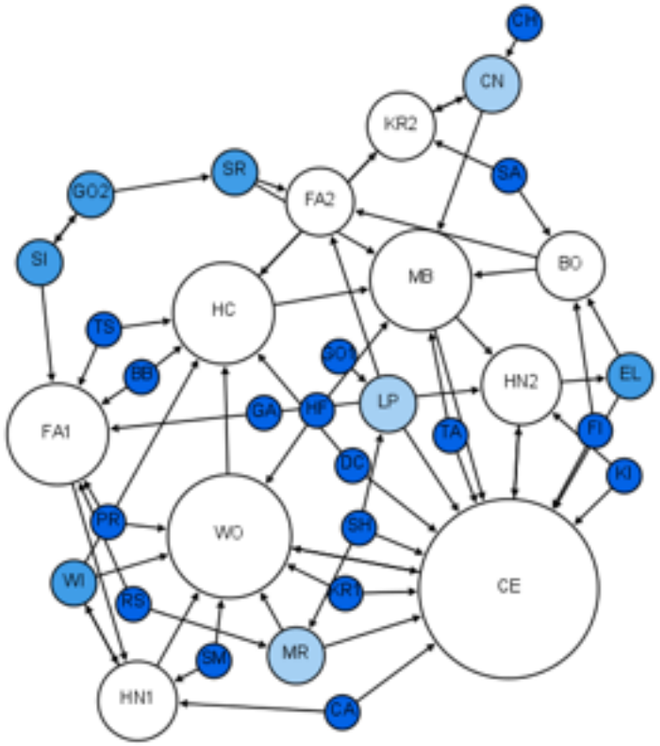
Source: Wikipedia

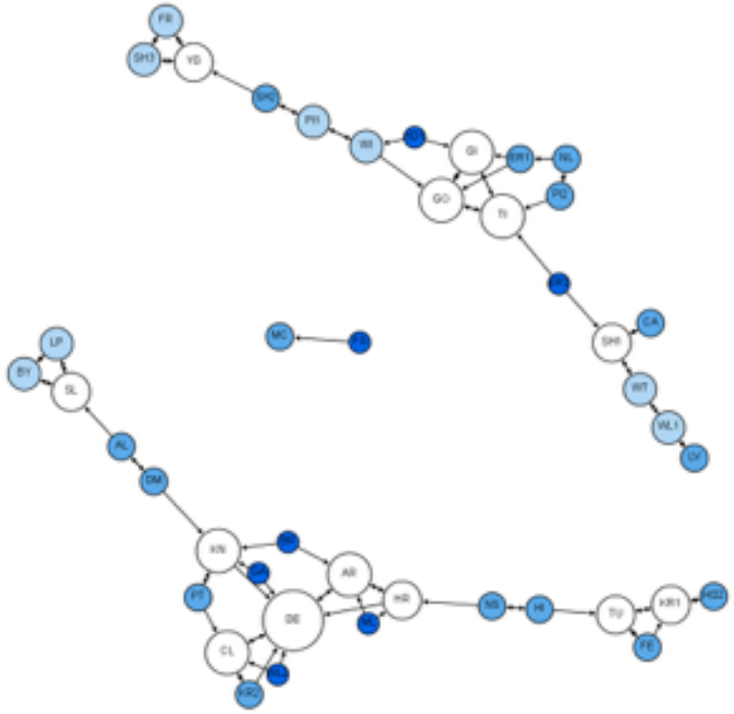
Solution: No! Unless a node is a starting or endpoint, it must have an even number of edges if every edge is traversed only once.

Social networks

- Jacob L. Moreno introduced sociograms in his 1934 book "Who Shall Survive?"
- Understand the individual through it's relation to the group







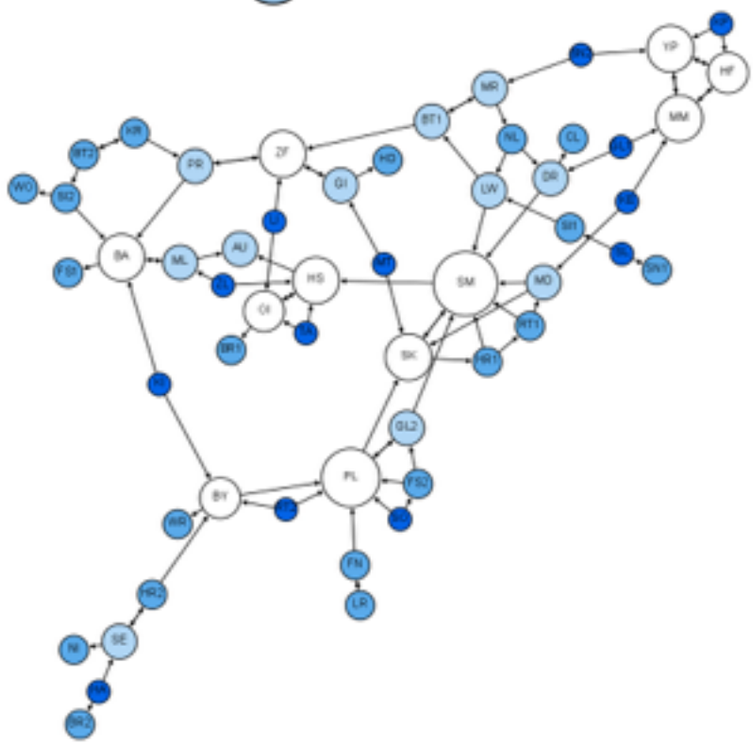


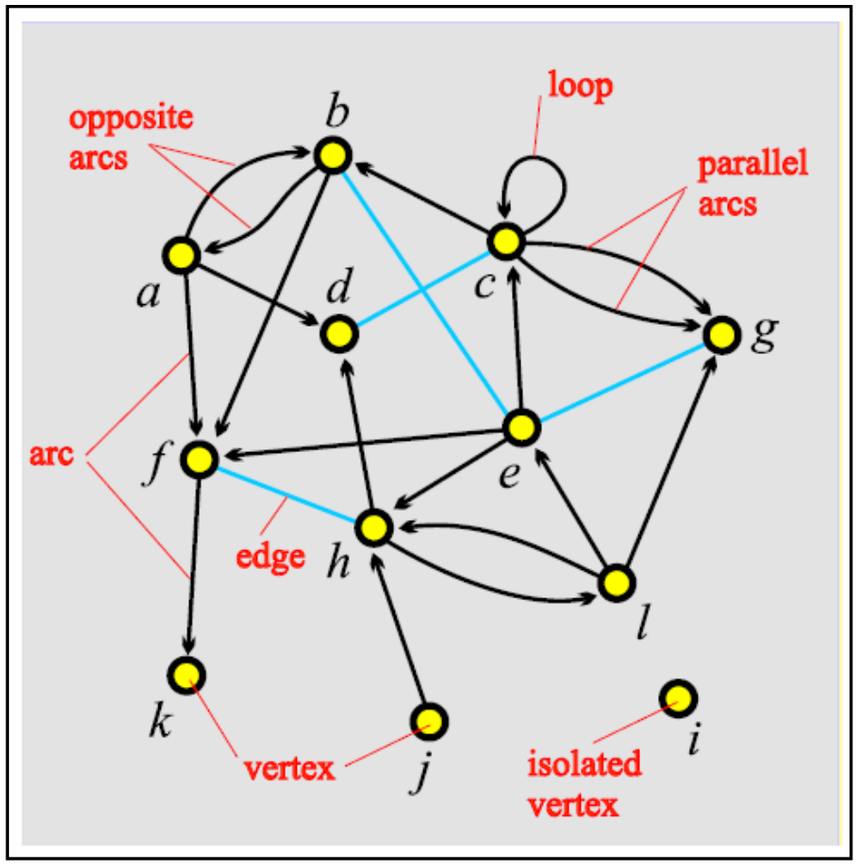
Table 3.1 *Basic statistics for a number of published networks. The properties measured are as follows: total number of vertices n; total number of edges m; mean degree z; mean vertex–vertex distance ℓ; type of graph, directed or undirected; exponent α of degree distribution if the distribution follows a power law (or "–" if not; in/out-degree exponents are given for directed graphs); clustering coefficient $C^{(1)}$ from* (3.3)*; clustering coefficient $C^{(2)}$ from* (3.6)*; degree correlation coefficient r, section* 3.6*. The last column gives the citation for the network in the bibliography. Blank entries indicate unavailable data.*

	Network	Type	n	m	z	ℓ	α	$C^{(1)}$	$C^{(2)}$	r	Ref(s).
Social	film actors	undirected	449 913	25 516 482	113.43	3.48	2.3	0.20	0.78	0.208	[20, 415]
	company directors	undirected	7 673	55 392	14.44	4.60	–	0.59	0.88	0.276	[105, 322]
	math coauthorship	undirected	253 339	496 489	3.92	7.57	–	0.15	0.34	0.120	[107, 181]
	physics coauthorship	undirected	52 909	245 300	9.27	6.19	–	0.45	0.56	0.363	[310, 312]
	biology coauthorship	undirected	1 520 251	11 803 064	15.53	4.92	–	0.088	0.60	0.127	[310, 312]
	telephone call graph	undirected	47 000 000	80 000 000	3.16		2.1				[8, 9]
	email messages	directed	59 912	86 300	1.44	4.95	1.5/2.0		0.16		[136]
	email address books	directed	16 881	57 029	3.38	5.22	–	0.17	0.13	0.092	[320]
	student relationships	undirected	573	477	1.66	16.01	–	0.005	0.001	−0.029	[45]
	sexual contacts	undirected	2 810				3.2				[264, 265]
Information	WWW `nd.edu`	directed	269 504	1 497 135	5.55	11.27	2.1/2.4	0.11	0.29	−0.067	[14, 34]
	WWW Altavista	directed	203 549 046	2 130 000 000	10.46	16.18	2.1/2.7				[74]
	citation network	directed	783 339	6 716 198	8.57		3.0/–				[350]
	Roget's Thesaurus	directed	1 022	5 103	4.99	4.87	–	0.13	0.15	0.157	[243]
	word co-occurrence	undirected	460 902	17 000 000	70.13		2.7		0.44		[119, 157]
Technological	Internet	undirected	10 697	31 992	5.98	3.31	2.5	0.035	0.39	−0.189	[86, 148]
	power grid	undirected	4 941	6 594	2.67	18.99	–	0.10	0.080	−0.003	[415]
	train routes	undirected	587	19 603	66.79	2.16	–		0.69	−0.033	[365]
	software packages	directed	1 439	1 723	1.20	2.42	1.6/1.4	0.070	0.082	−0.016	[317]
	software classes	directed	1 377	2 213	1.61	1.51	–	0.033	0.012	−0.119	[394]
	electronic circuits	undirected	24 097	53 248	4.34	11.05	3.0	0.010	0.030	−0.154	[155]
	peer-to-peer network	undirected	880	1 296	1.47	4.28	2.1	0.012	0.011	−0.366	[6, 353]
Biological	metabolic network	undirected	765	3 686	9.64	2.56	2.2	0.090	0.67	−0.240	[213]
	protein interactions	undirected	2 115	2 240	2.12	6.80	2.4	0.072	0.071	−0.156	[211]
	marine food web	directed	135	598	4.43	2.05	–	0.16	0.23	−0.263	[203]
	freshwater food web	directed	92	997	10.84	1.90	–	0.40	0.48	−0.326	[271]
	neural network	directed	307	2 359	7.68	3.97	–	0.18	0.28	−0.226	[415, 420]

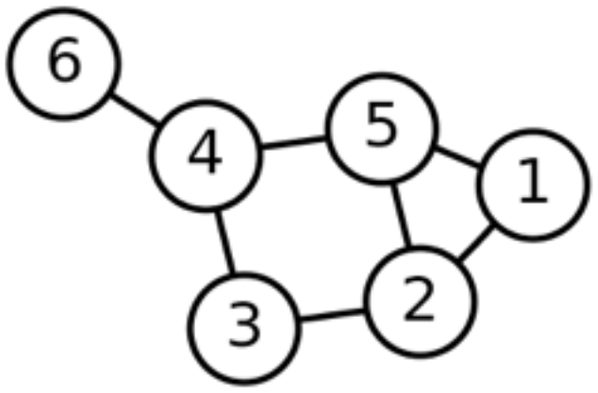
Source: Newman, M.E. (2003). *The structure and function of complex networks*. SIAM review, 45(2), 167-256.

Basic definitions

- A graph *G* is defined as *G(N,L)*
  - Set of nodes (vertices) *N*
    - Nodes can have **attributes**
- Set of links (edges) *L*
  - **Directed** (arcs) or **undirected** (edges)
  - **Unweighted** or **weighted** (distance, traveling time, etc.)
  - Links of different types can exist (multiplex networks)



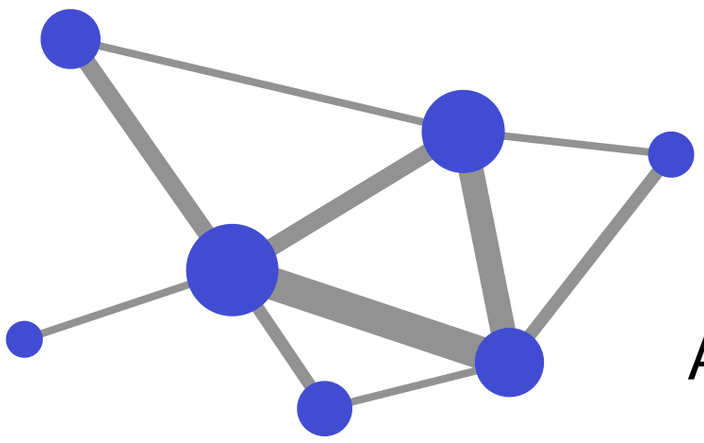
Network Representations: Adjacency Matrix



a_{ij} = existence of interaction between *i* and *j*

w_{ij} = weights of interaction between *i* and *j*
(e.g. number of communications per unit time)

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$



Adjacency Matrix **A** has entries a_{ij}

$$a_{ij} = \begin{cases} 1, & \text{if } w_{ij} > 0 \\ 0, & \text{else} \end{cases}$$

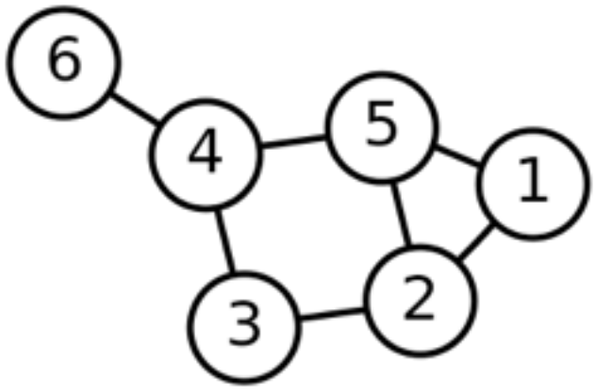
Network Representations: Edge and Adjacency Lists

Edge list

1	2
1	5
2	1
2	5
3	2
3	4
4	3
4	5
4	6
5	1
5	2
5	4
6	4

Adjacency list

1	2 5
2	1 5
3	2 4
4	3 5 6
5	1 2 4
6	4

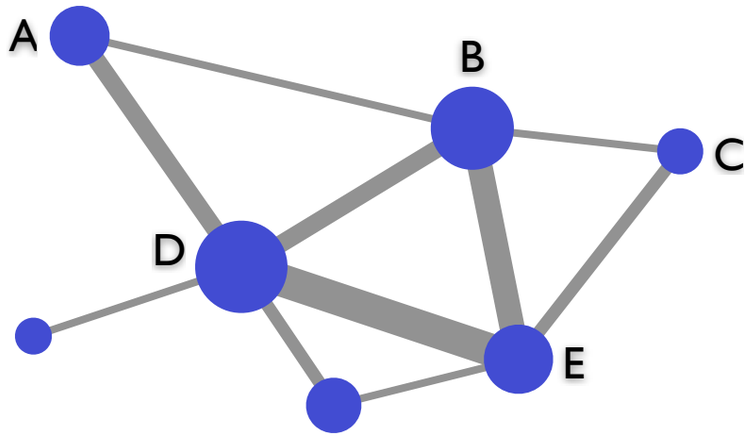


$$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Paths

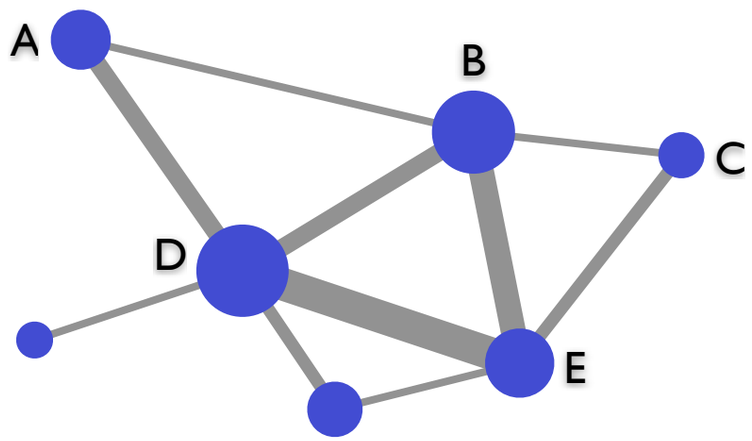
- **Path** of length n = ordered collection of n+1 nodes and n links.
  - Eg: (*A,B,C,E),* (*A,D*), (*C,D*),(*D,E,C*) in *G* =(*N,L*)
- **Circuit** = closed path (last node = first node)

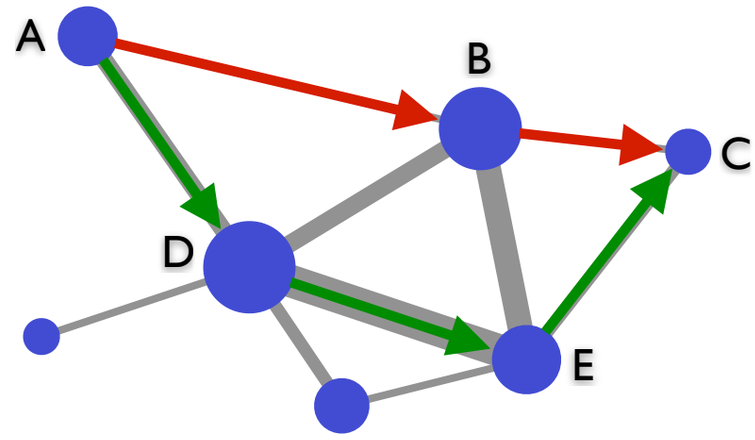
Number of **walks** length *k* is given by powers or adjacency matrix



Geodesic paths

- The geodesic (shortest) path between *i* and *j* is minimum number of traversed edges

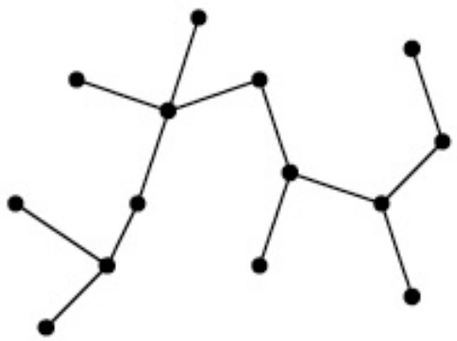




- Distance *d(i,j)* = shortest path between *i* and *j*
- Diameter *D* of the graph = max(*d(i,j)*)

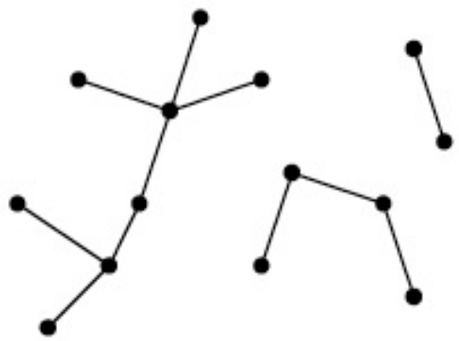
Connected components

- A graph *G=(N,L)* is connected if and only if there exists a path connecting any two nodes in *G*



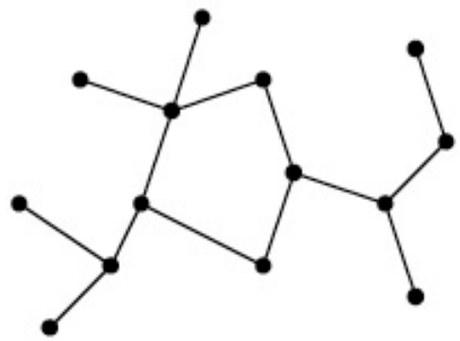
(a)

Connected (Tree)



(b)

Not Connected (Forest)

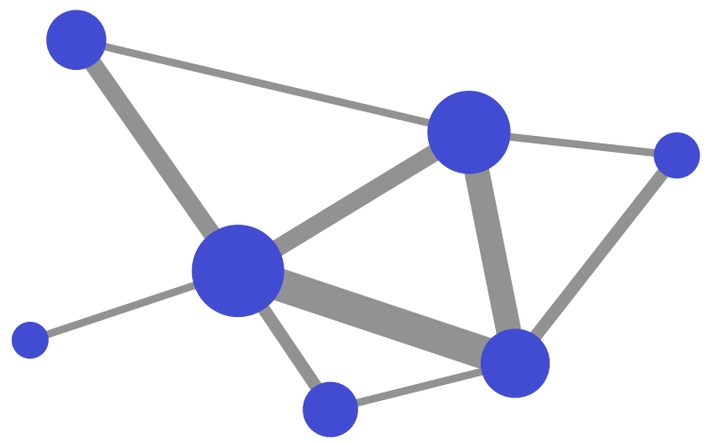


(c)

Connected with loops

Centrality Measures

Degree, Strength, Closeness



a_{ij} = existence of interaction between *i* and *j*

w_{ij} = weight of interaction between *i* and *j (*e.g. number of communications per unit time)

d_{ij} = distance between *i* and *j*

Node degree

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

Node flux/strength

$$F_i = \sum_j w_{ij}$$

Node closeness

$$D_i = \sum_j d_{ij}$$

Eigenvector centrality and PageRank

Brin, Sergey, and Lawrence Page. "Reprint of: The anatomy of a large-scale hypertextual web search engine." Computer networks 56.18 (2012): 3825-3833.

- **Eigenvector centrality** x_i is higher the more high-scoring others a node is connected to:

$$x_i^{(t+1)} = \sum_{j=1}^{n} A_{ij} x_j^{(t)}$$

Solution is dominated by the largest eigenvalue λ_1 as $t \to \infty$

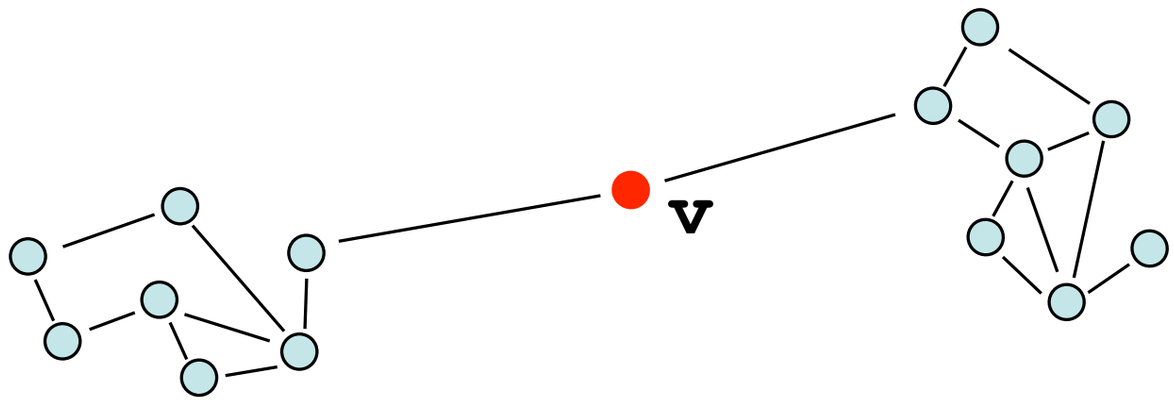
$$\mathbf{Ax} = \lambda_1 \mathbf{x} \qquad x_i = \lambda_1^{-1} \sum_{j=1}^{n} A_{ij} x_j$$

- **PageRank** x_i downgrades common in-links and deals with directed links:

$$x_i = \alpha \sum_{j=1}^{n} A_{ij} \frac{x_j}{k_j^{\text{out}}} + \beta \qquad \mathbf{x} = \alpha \mathbf{A} \mathbf{D}^{-1} \mathbf{x} + \beta \mathbf{1}$$

$$\mathbf{D} = max(k_{\text{out}}, 1)$$

Betweenness centrality



- Idea: Controlling network flows
- The number of shortest paths passing through a node v:

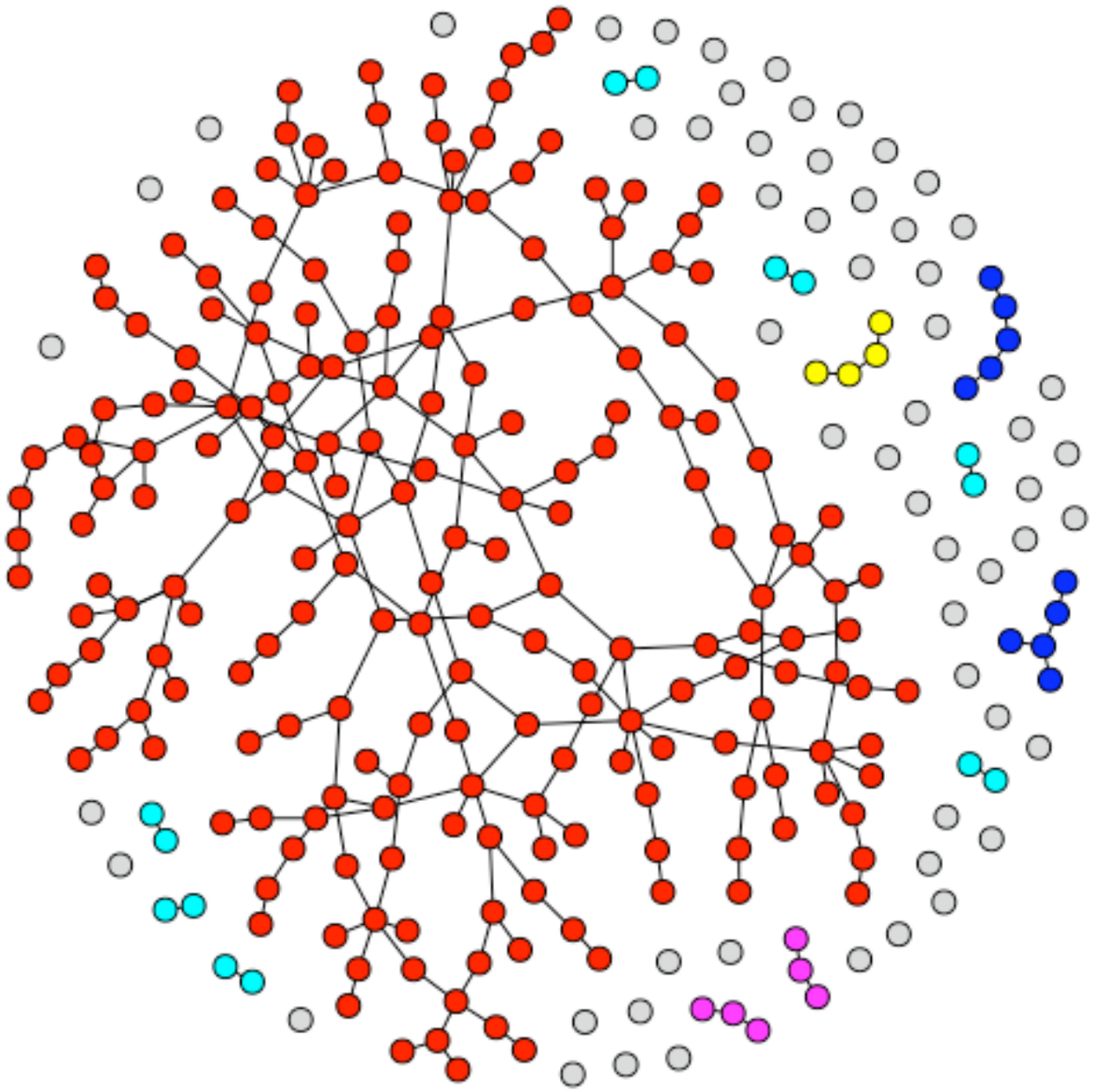
$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

σ_{st} = number of shortest paths from s to t

$\sigma_{st}(v)$ = number of shortest paths from s to t passing through v

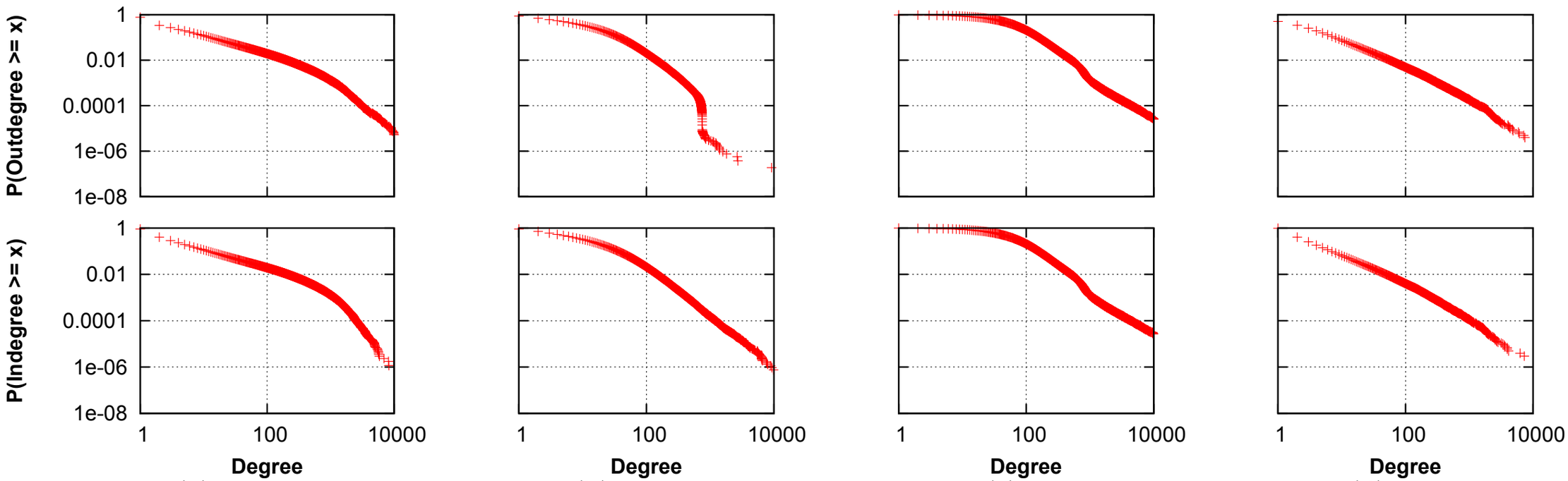
Structural features (of social networks)

Giant Component



A giant component is a connected component which size scales with the size of the network

Centrality heterogeneity

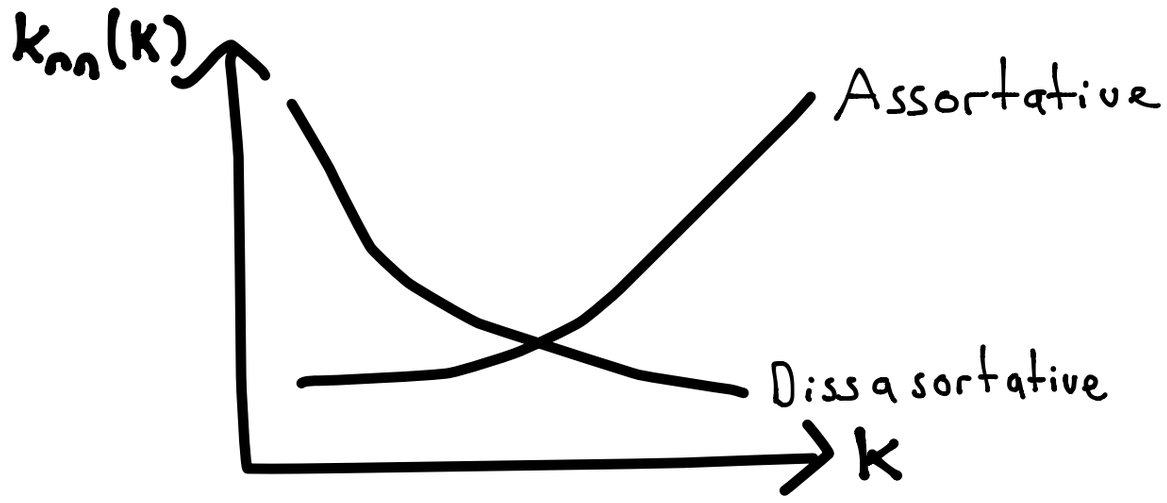


(a) Flickr (b) LiveJournal (c) Orkut (d) YouTube

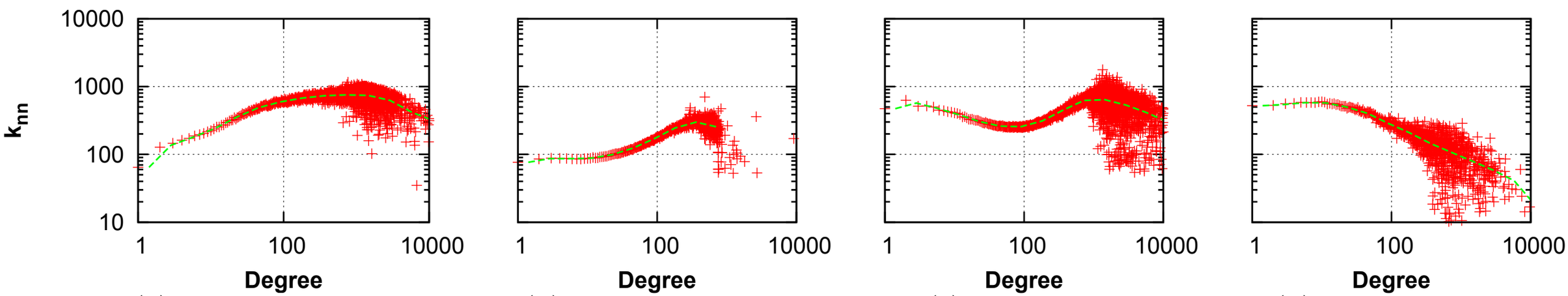
source: Mislove et al. (2007)

Assortative mixing or homophily

- Birds of a feather flock together
- Can be any characteristic
- E.g. Degree assortativity:



- Average nearest-neighbor degree for vertices with degree *k*

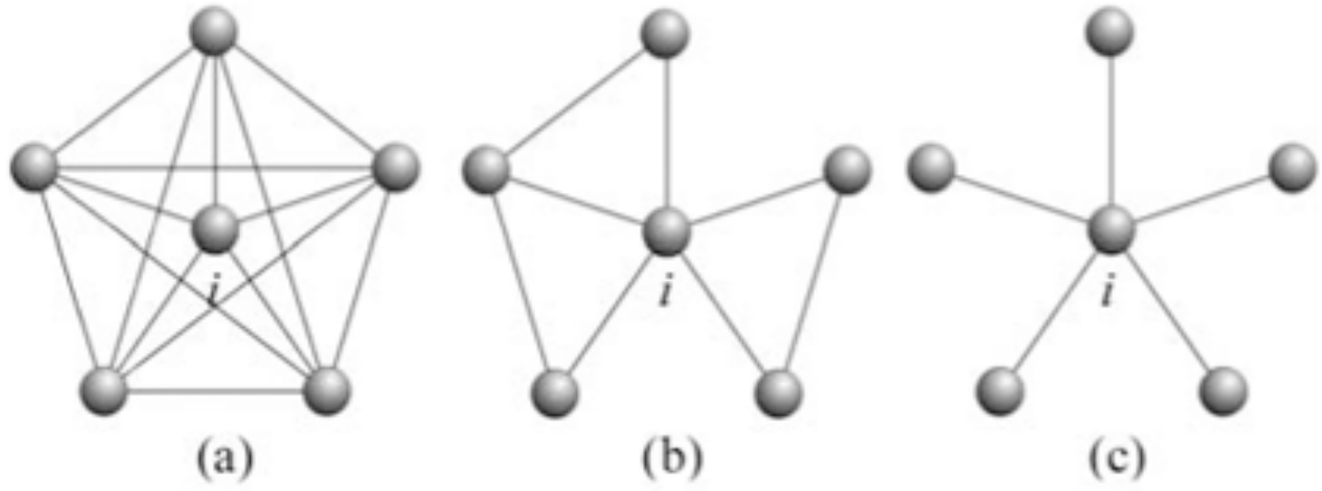


(a) Flickr (b) LiveJournal (c) Orkut (d) YouTube

source: Mislove et al. (2007)

Transitivity and clustering

- My friends tend to be friends
- **Local clustering coefficient C(i)**: fraction of pairs of neighbors of a node that are also neighbors of each other. Equivalently, number of closed triples.

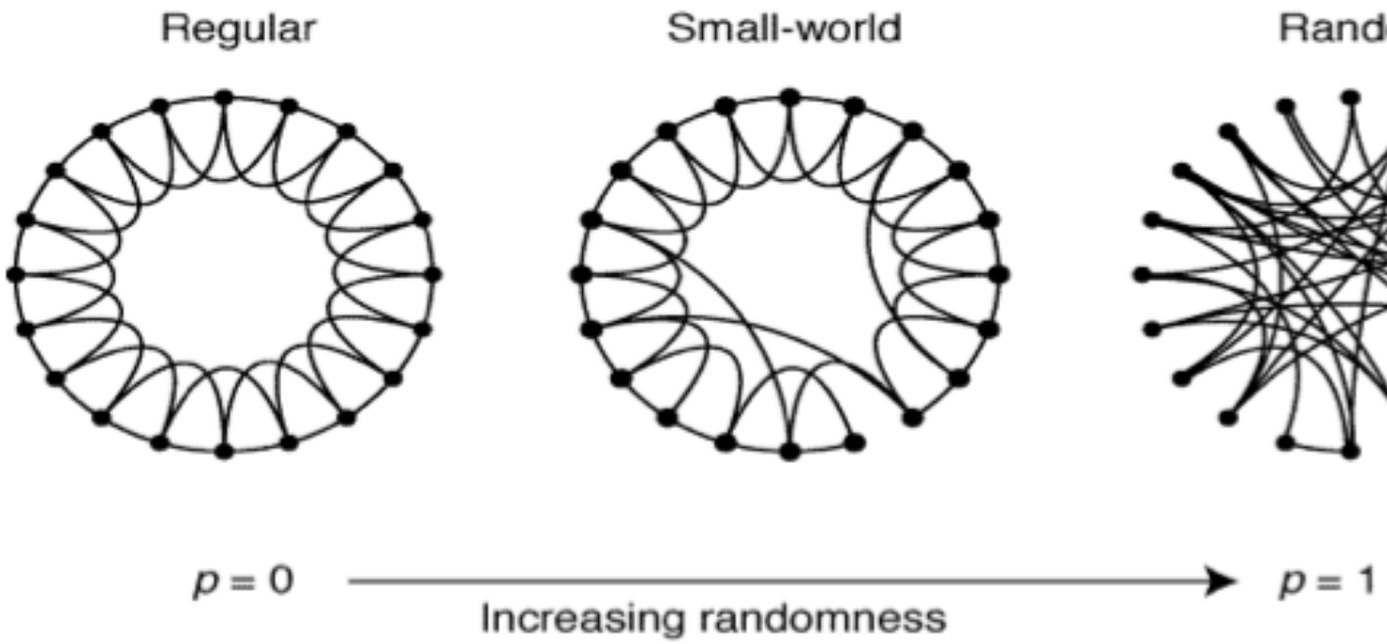


Source Costa (2008)

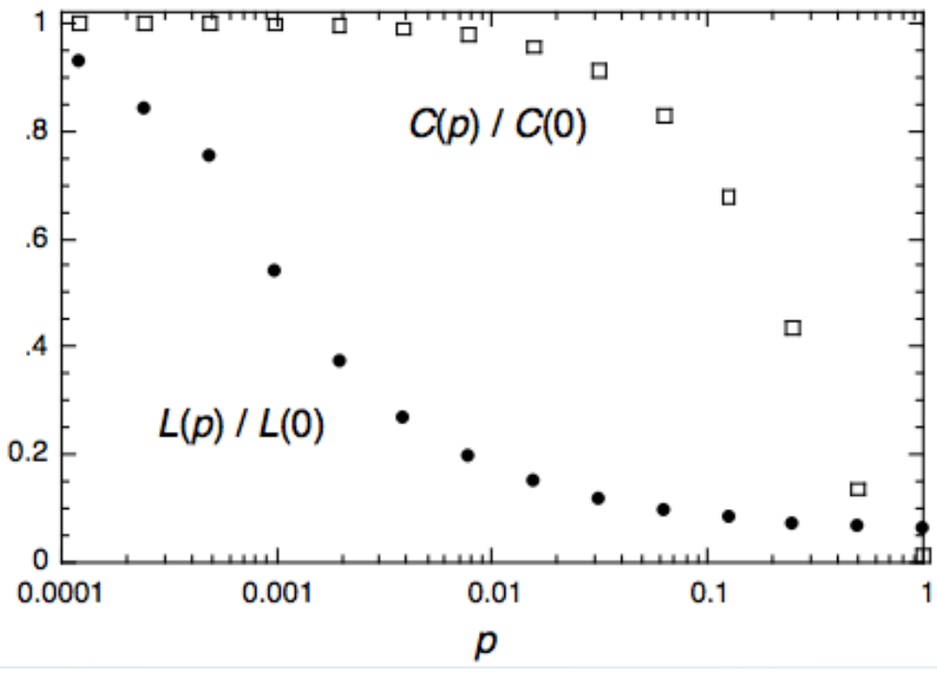
- Question: What is the local clustering coefficient for the node *i* ?
- **Global clustering coefficient**: network average

Small world property

- Empirical puzzle: Social worlds that are highly clustered but at the same time global distances are short — e.g. there are at most 6 degrees of separation between any two randomly chosen individuals



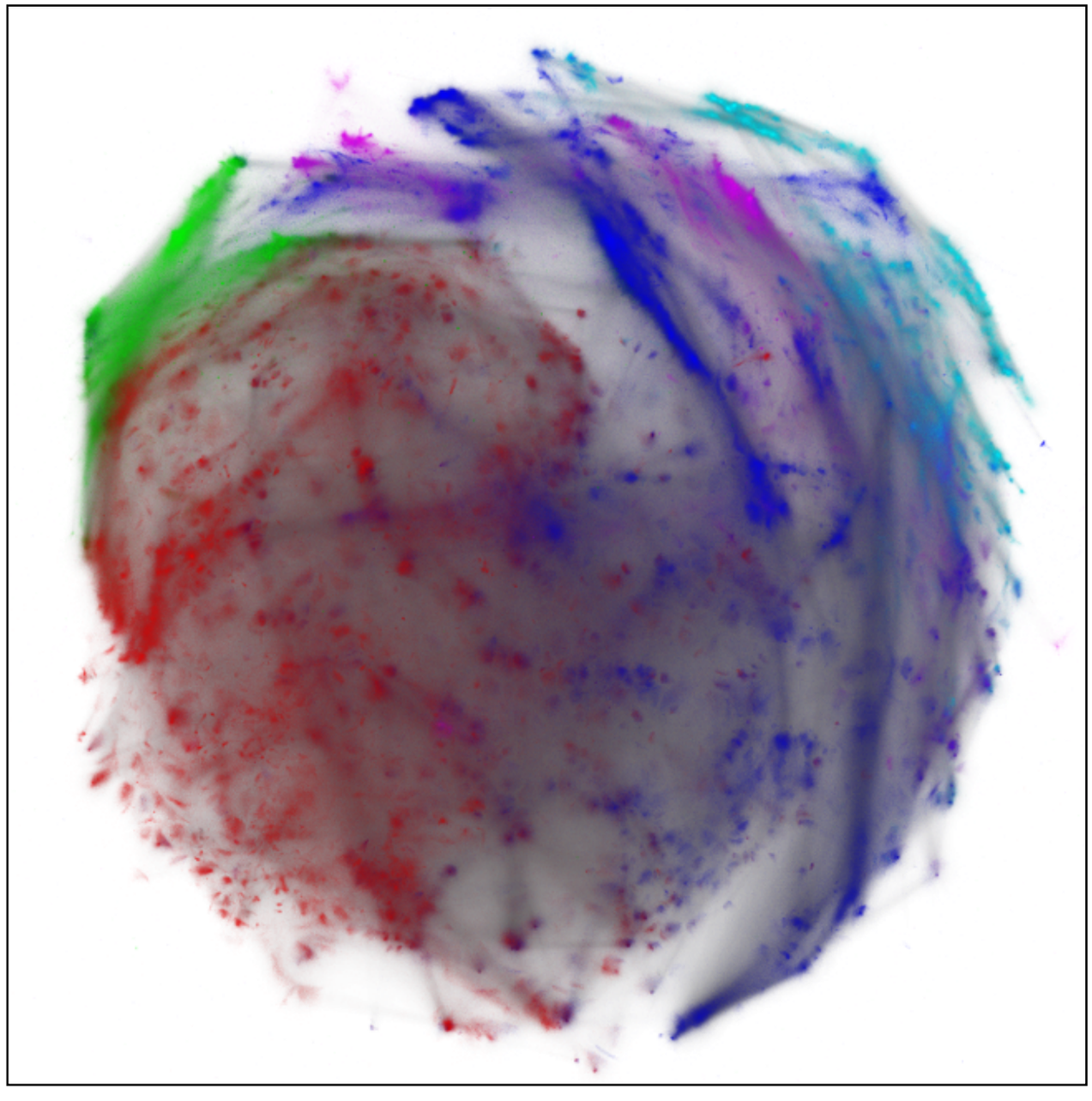
Randomly rewire a link with probability p

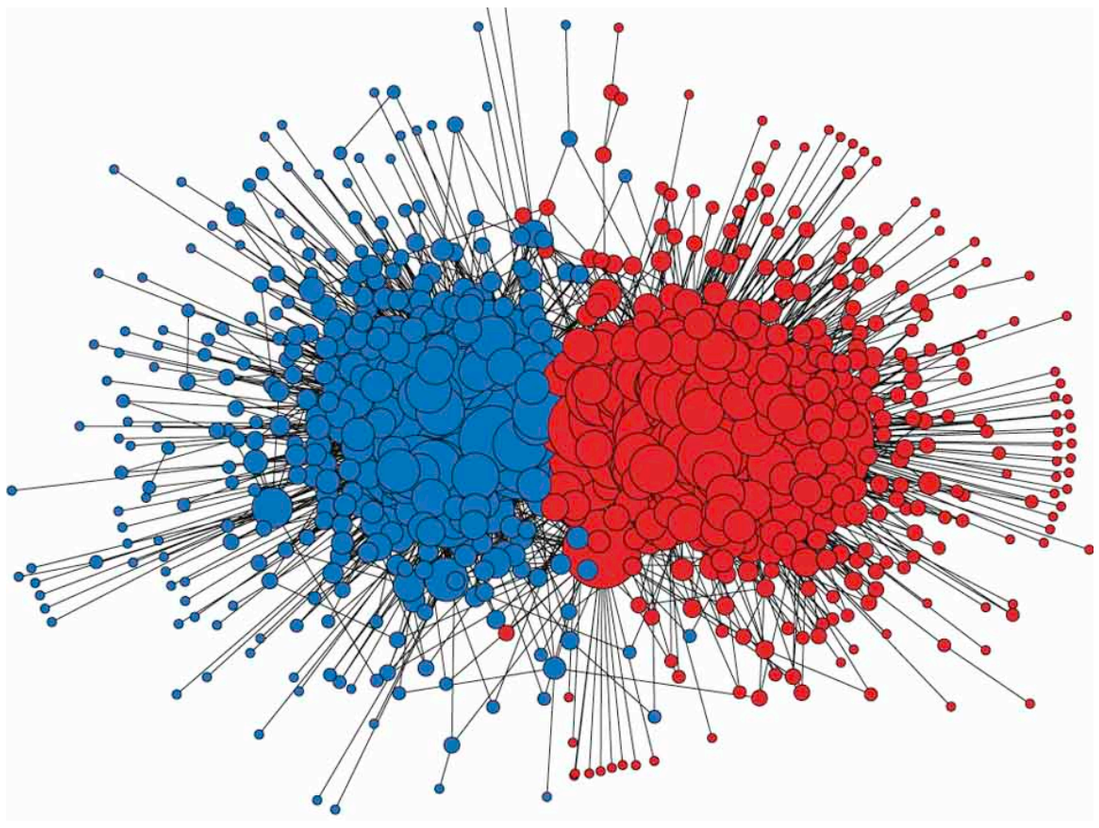


Source: Watts, D. J., & Strogatz, S. H. (1998)

- A small-world network is a network where the typical distance L between two randomly chosen nodes grows logarithmically with total number of nodes N

Modularity and community structure





Source: Newman, M. E. J. (2011)

Community detection vs Graph partition

- **Graph partitioning** specifies the number of subgroups or number of nodes in each subgroup
    - Hierarchical clustering, *k* means
- **Community detection** infers the subgroups from the network structure
    - Divisive algorithms (recursively removing highest betweenness edges)
    - Random walk algorithms (maximizing the time random walkers spend within a community)
    - Modularity

Modularity

$$Q = \sum_{s=1}^{m} \left[\frac{l_s}{L} - \left(\frac{d_s}{2L} \right)^2 \right]$$

observed fraction within group connections

expected fraction within group connections

m modules
L links in total
l_s links within module s
d_s total module degree

- Basic idea: High fraction of links within group compared to chance (a null model)
- Community detection: Find partition with maximal modularity *Q*

Network Models

1. Random graphs
2. Configuration models
3. Generative models

1. Random graphs (Erdos-Renyi)

- Start with a number of nodes *n* (not connected)
- Define probability of connection *p*
- For all the possible couples of nodes a link is created with probability *p*

Degree and clustering are easily computable

- The degree distribution is the Binomial distribution

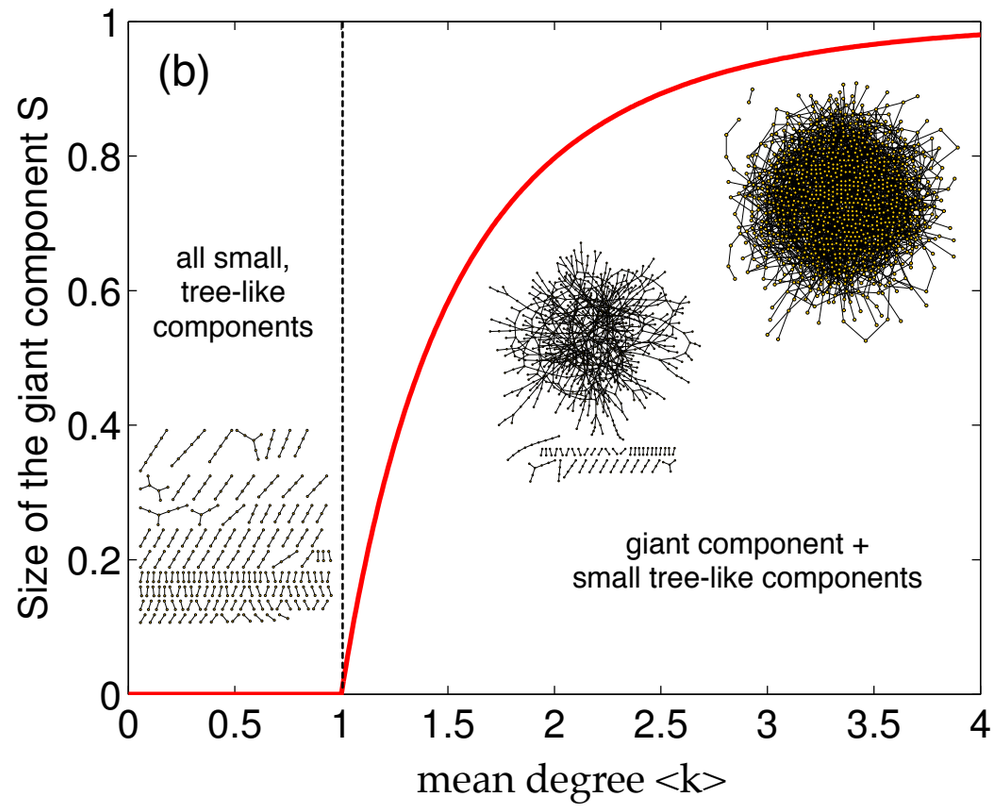
$$\Pr(k) = \binom{n-1}{k} p^k (1-p)^{n-1-k}$$

in the limit of large *n* $\Pr(k) \simeq \frac{(n-1)^k}{k!} p^k \mathrm{e}^{-c} = \frac{c^k}{k!} \mathrm{e}^{-c}$

- The average degree is: <*k*> = *p(n-1)*
- Clustering coefficient *C* is simply *p* (the probability of any pair existing)
- No heterogeneous degree distributions
- No small-world scaling with clustering

Percolation transition

- The formation of the Giant Component is not a smooth process
- Emerges suddenly when *<k>=1*
- This phenomenon is called 1st order phase-transition



Source: A. Clauset Network lectures
http://tuvalu.santafe.edu/~aaronc/courses/5352/

2. Configuration model

- Fix the degree sequence or degree distribution
- Find a network that samples uniformly over all other properties
- E.g. assign degrees to nodes and add “stubs”

- Uniformly at random sample stubs and connect them
- Problem : Creates self and duplicate edges (works better as network size grows)
- This process can be generalized to any property (see also Exponential Random Graph models)

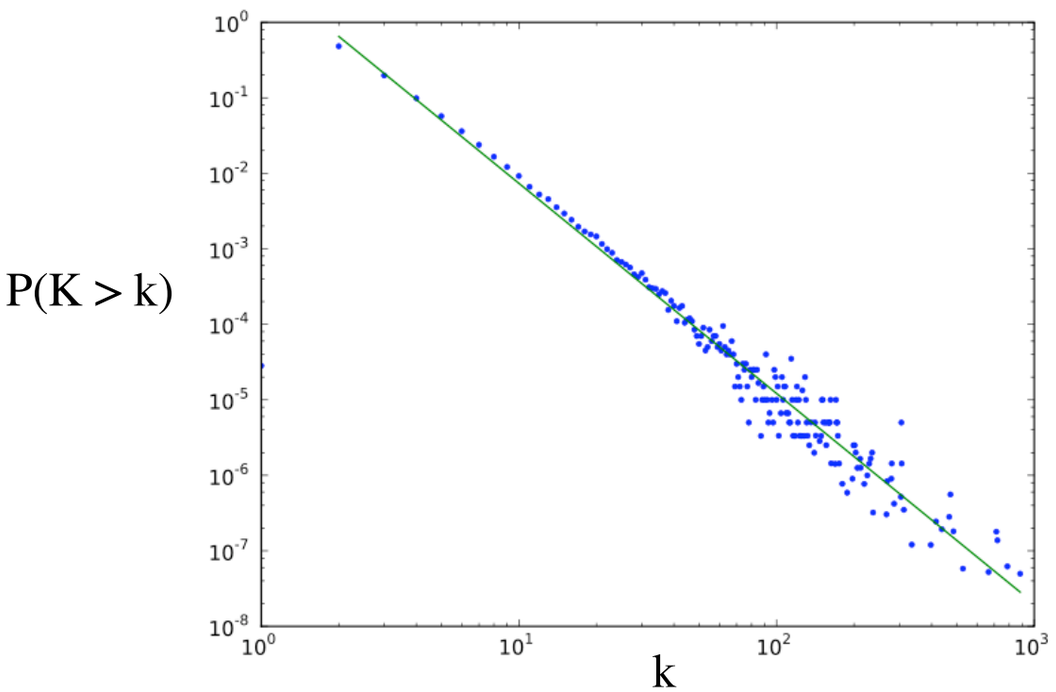
3. Generative models: Preferential attachment

- Algorithm:
  - Start with a random connected graph
  - At each time step create a new node and attach it to each node *i* with probability p_i proportional to the node degree k_i

$$p_i = \frac{k_i}{\sum_j k_j}$$

- Generates power-law tails (richer-get-richer)

$$P(K) \sim k^{-3}$$



Network packages

- MATLAB: MatlabBGL (Boost Graph Library) http://www.mathworks.com/matlabcentral/fileexchange/10922-matlabbgl or http://dgleich.github.io/matlab-bgl/
- Python: NetworkX https://networkx.github.io/
- iGRAPH http://igraph.org/redirect.html (originally R, now also python and C/C++)

Representing and visualizing networks

- Gephi (http://gephi.org) -> Easy and common
- Pajek (http://pajek.imfm.si/doku.php) -> Easy to use
- NWB (http://nwb.cns.iu.edu) -> Good for Analysis
- Visone (http://visone.info)
- JUNG (http://jung.sourceforge.net) -> library
- Net Draw (http://www.analytictech.com/netdraw/netdraw.htm)
- Pegasus (http://www.cs.cmu.edu/~pegasus) -> for huge data

References

- Handbook of graphs and networks: from the Genome to the Internet, edited by S. Bornholdt, H. G. Schuster. John Wiley and Sons, 2003.
- Watts,D.J.,& Strogatz, S.H. (1998).Collective dynamics of ‘small- world’ networks. nature, 393(6684), 440-442.
- Newman, M.E. (2003).The structure and function of complex networks. SIAM review, 45(2), 167-256.
- Mislove, A., et al. (2007) "Measurement and analysis of online social networks." Proceedings of the 7th ACM SIGCOMM conference on Internet measurement. ACM.
- Newman, M. E. (2009). Networks: an introduction. Oxford University Press.
- Easley, D., & Kleinberg, J. (2010). Networks, crowds, and markets. Cambridge: Cambridge University Press.
- Newman, M. E. J. (2011). Communities, modules and large-scale structure in networks. Nature Physics, 8(1), 25-31.
- Fortunato, S. "Community detection in graphs." Physics reports 486.3 (2010): 75-174.
- Laszlo Barabasi web site: http://nd.edu/~alb/